

---

Università degli Studi di Napoli Federico II



Dottorato in Ingegneria Aerospaziale, Navale e della Qualità  
XXIV ciclo

Dipartimento di Ingegneria Aerospaziale

---

# Aerodynamic Shape Optimization with Physics-based Surrogate Models

## **Tutor**

Ch.mo prof. Carlo de Nicola  
Ch.mo dott. Domenico Quagliarella

## **Candidato**

Emiliano Iuliano

ANNO ACCADEMICO 2011

## Abstract

Aircraft design, as many other engineering applications, are increasingly relying on computational power. The growing need for multi-disciplinarity and high-fidelity in design optimization and industrial applications implies a huge number of repeated simulations to find an optimal design candidate. Indeed, a strong effort has been done in the recent past to introduce potentially highly accurate analysis methods both in geometry and physics modelling. The main drawback is that they are computationally expensive. The solution of non-linear steady or unsteady aerodynamic flows by numerically solving the Navier-Stokes equations implies an amount of data storage, data handling and processor costs that may result very intensive even when implemented on modern state-of-art computing platforms. This turns out to be an even bigger issue when used within parametric studies, automated search or optimization loops which typically may require thousands analysis evaluations. The core issue of a design optimization problem is the search process of an optimal solution. However, when facing complex problems, the high-dimensionality of the design space and the high-multi-modality of the target functions cannot be tackled with standard techniques. Surrogate and reduced order modelling can provide a valuable alternative at a much lower computational cost. A global surrogate model is generally referred to as a low-cost model able to provide an approximation of a selected objective function over the whole design space. A reduced order model is a surrogate which is further able to capture and reproduce the physics embedded in the high-fidelity model by using a low-dimensional basis. Hence, a reduced order modelling of high-fidelity data (e.g. coming from accurate numerical solvers) with limited computational cost is a highly desirable feature. This is particularly true in CFD-based aerodynamic optimization. Commonly used RANS solvers are still time-consuming when complex fluid dynamics cases

have to be faced, e.g. a wing-body aircraft configuration including engine and tailplanes. In this perspective, the present research aims at making a step towards bridging the gap between design stages through the coupling and exploitation of advanced analysis methods, reduced order/meta-modeling, optimization techniques and CAD-based tools towards the aerodynamic design of innovative aircraft configurations with reasonable computational resources. The introduction of physics-based surrogate models will allow to correctly drive the design process since the very early stages and, hence, to refine the evaluation of potentially cost/environment saving concepts.





# Contents

<b>List of Figures</b>	<b>iii</b>
<b>List of Tables</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivations . . . . .	1
1.2 Surrogate-Based Optimization . . . . .	3
1.2.1 The Aircraft Design Optimization Problem . . . . .	5
1.2.1.1 The importance of geometry parameterization . . . . .	7
1.2.2 Surrogate models . . . . .	8
1.2.2.1 Data-fit models . . . . .	8
1.2.2.2 Multi-fidelity models . . . . .	9
1.2.2.3 Reduced order modelling . . . . .	9
1.2.2.4 An example of dimensionality reduction . . . . .	10
1.3 Aim and objectives . . . . .	14
1.4 Thesis outline . . . . .	15
<b>2 Literature review</b>	<b>17</b>
2.1 Surrogate-based Optimization . . . . .	17
2.2 Proper Orthogonal Decomposition . . . . .	25
2.3 Progress beyond the State of the Art . . . . .	33
2.3.1 Progress in geometry parameterization . . . . .	33
2.3.2 Progress in POD-based surrogate modelling . . . . .	34
2.3.3 Progress in transonic flow shape optimization . . . . .	34

<b>3</b>	<b>Ingredients for aerodynamic design optimization</b>	<b>37</b>
3.1	An evolutionary optimization tool for aerodynamic design . . . . .	38
3.1.1	Optimization and design concepts . . . . .	39
3.1.2	Problem definition . . . . .	41
3.1.3	Configuration definition and geometry parameterization . . . . .	41
3.1.3.1	The Class-Shape Transformation . . . . .	43
3.1.3.2	Airfoil representation . . . . .	43
3.1.3.3	Wing representation . . . . .	52
3.1.3.4	The integration within SALOME GEOM tool . . . . .	53
3.1.4	Aerodynamic analysis . . . . .	55
3.1.4.1	Governing equations . . . . .	57
3.1.4.2	Reynolds Averaging of the Navier-Stokes Equations . . . . .	60
3.1.4.3	The ZEN flow solver . . . . .	62
3.1.5	Genetic algorithm - The ADGLIB library . . . . .	65
3.2	A state of the art optimization study . . . . .	68
3.2.1	Problem Definition . . . . .	69
3.2.2	Optimization strategy . . . . .	70
3.2.3	Wing section optimization with a 2.5D approach . . . . .	71
3.2.4	Automatic 3D mesh generation . . . . .	78
3.2.5	Wing-body-nacelle optimization . . . . .	79
3.2.5.1	Baseline analysis . . . . .	79
3.2.5.2	Discussion of optimization results . . . . .	80
3.2.6	Concluding remarks . . . . .	87
<b>4</b>	<b>The Proper Orthogonal Decomposition</b>	<b>91</b>
4.1	Reduced order models . . . . .	91
4.2	POD theory . . . . .	93
4.3	POD solution . . . . .	96
4.3.1	Snapshots collection . . . . .	96
4.3.2	SVD solution . . . . .	97
4.3.3	Pseudo-continuous global representation . . . . .	98

<b>5</b>	<b>POD-based Reduced Order Modelling for transonic shape optimization</b>	<b>103</b>
5.1	Design space definition . . . . .	103
5.2	POD/ROM with varying geometry and grid . . . . .	105
5.3	Design of Experiments . . . . .	105
5.3.1	Latin Hypercube sampling . . . . .	106
5.4	Zonal POD . . . . .	107
5.5	Error analysis . . . . .	110
5.6	Validation of POD/ROMs in transonic flow . . . . .	111
5.7	Adaptive sampling . . . . .	120
5.7.1	Improvement of the modal basis . . . . .	124
5.7.2	Improvement of the coefficients models . . . . .	126
5.7.2.1	First strategy . . . . .	126
5.7.2.2	Second strategy . . . . .	128
5.8	Concluding remarks . . . . .	128
<b>6</b>	<b>POD-based Evolutionary Optimization in Transonic flow</b>	<b>131</b>
6.1	The surrogate-based shape optimization framework . . . . .	131
6.2	Problem definition . . . . .	132
6.3	Optimization strategies and set up . . . . .	134
6.4	Non-adaptive optimization results . . . . .	139
6.5	Adaptive optimization results . . . . .	139
6.6	Optima analysis . . . . .	144
6.6.1	Non-adaptive optima . . . . .	144
6.6.2	Adaptive optima . . . . .	146
<b>7</b>	<b>Conclusions</b>	<b>155</b>
	<b>References</b>	<b>157</b>



# List of Figures

1.1	Aerodynamic drag reduction technology . . . . .	2
1.2	Three-dimensional data set, graphical representation . . . . .	11
1.3	POD basis graphical representation . . . . .	12
1.4	3D data reconstruction with 2 POD modes - XY plane projection . . .	14
2.1	DACE prediction error . . . . .	20
2.2	Expected Improvement measure to improve the goodness-of-fit . . . . .	20
2.3	Taxonomy of response surface-based global optimization methods, according to Jones (1) . . . . .	21
2.4	SBO methods taxonomy according to Forrester . . . . .	24
2.5	Example of SBO minimizing the predictor . . . . .	26
3.1	NACA-type airfoil, $N_1=0.5$ , $N_2=1.0$ . . . . .	44
3.2	Elliptic airfoil, $N_1=0.5$ , $N_2=0.5$ . . . . .	44
3.3	Biconvex airfoil, $N_1=1.0$ , $N_2=1.0$ . . . . .	44
3.4	Duct shape, $N_1=0.001$ , $N_2=0.001$ . . . . .	44
3.5	Low-drag projectile, $N_1=0.75$ , $N_2=0.25$ . . . . .	44
3.6	Wedge airfoil, $N_1=1.0$ , $N_2=0.001$ . . . . .	44
3.7	Decomposition of the shape function in two component airfoils . . . . .	45
3.8	Bernstein component shapes . . . . .	46
3.9	Test shapes . . . . .	48
3.10	Maximum residual error $RM$ as a function of design variables number .	49
3.11	Residual error sum of squares $RSS$ as a function of design variables number	49
3.12	Airfoil curvature analysis . . . . .	50
3.13	Pressure coefficient analysis . . . . .	50

3.14	Skin friction analysis . . . . .	51
3.15	Bi-variate Bernstein polynomials . . . . .	54
3.16	Wing shaping within SALOME . . . . .	56
3.17	Smooth transition of randomly assigned wing sections along the span . .	56
3.18	Sectional distributions along the wing span . . . . .	72
3.19	2.5D optimization convergence histories . . . . .	74
3.20	Wing section polar curves at cruise conditions . . . . .	75
3.21	Transition prediction as a function of $C_l$ . . . . .	76
3.22	Airfoil optimization results . . . . .	77
3.23	The aircraft configuration . . . . .	78
3.24	RANS skin mesh with block structure highlight . . . . .	79
3.25	Baseline wing analysis . . . . .	81
3.26	Baseline wing analysis . . . . .	82
3.27	GA convergence history . . . . .	84
3.28	Optimal wing analysis . . . . .	85
3.29	Optimal wing pressure distribution . . . . .	86
3.30	Transition lines on pressure contours at various AOAs . . . . .	88
3.31	% Laminar extent comparison . . . . .	88
3.32	Polar curves comparison . . . . .	89
5.1	Baseline geometry, RAE 2822 airfoil . . . . .	104
5.2	Standard LHS distribution of points . . . . .	107
5.3	Zonal approach, FOM/ROM domains . . . . .	108
5.4	Boundary condition setting . . . . .	109
5.5	180-sized snapshot airfoils . . . . .	112
5.6	180-sized snapshots ensemble . . . . .	112
5.7	Volume mesh around the airfoil . . . . .	113
5.8	FOM/ROM domains with varying interface . . . . .	114
5.9	Effect of zonal interface on the energy amount captured by POD . . . .	116
5.10	Surrogate models predictions of $C_L/C_D$ on the validation plan . . . . .	118
5.11	Pressure coefficient comparison . . . . .	121
5.12	Skin friction coefficient comparison . . . . .	122
5.13	Snapshot error prediction . . . . .	123

5.14	Trade-off accuracy vs resource saving . . . . .	124
6.1	Workflow of CFD/POD-based genetic optimization . . . . .	133
6.2	Non-adaptive POD-driven optimization history . . . . .	140
6.3	Kriging-based optimization history . . . . .	141
6.4	EGO optimization convergence history . . . . .	142
6.5	AFPGA optimizations convergence history . . . . .	143
6.6	MFPGA optimizations convergence history . . . . .	143
6.7	Non-adaptive optimal candidates comparison . . . . .	145
6.8	MPGA1 optimum, velocity contour comparison (POD - dashed lines; CFD - solid lines) . . . . .	145
6.9	Computed optima in the surrogate vs truth objective plane . . . . .	148
6.10	Optimal candidates, Mach number contour . . . . .	149
6.11	Optimal candidates, Mach number contour . . . . .	150
6.12	Optimal candidates, pressure coefficient distributions . . . . .	151
6.13	Optimal candidates, pressure coefficient distributions . . . . .	152
6.14	Optimal candidates, skin friction distributions . . . . .	153
6.15	Optimal candidates, skin friction distributions . . . . .	154





# List of Tables

1.1	The aircraft design phases according to Raymer . . . . .	5
1.2	Three-dimensional data set . . . . .	10
3.1	Clean aircraft operating points . . . . .	69
3.2	Optimization problem definition . . . . .	70
3.3	3D Optimization problem definition . . . . .	83
5.1	RAE 2822, 7 <sup>th</sup> -order Bernstein polynomials design weights . . . . .	104
5.2	Design point . . . . .	113
5.3	Surrogate goodness-of-fit estimation . . . . .	117
5.4	Surrogate estimations of aerodynamic efficiency for best and worst validation airfoils . . . . .	119
6.1	Optimization approaches . . . . .	135
6.2	Optimal candidates, obj. function breakdown . . . . .	144



# 1

## Introduction

### 1.1 Motivations

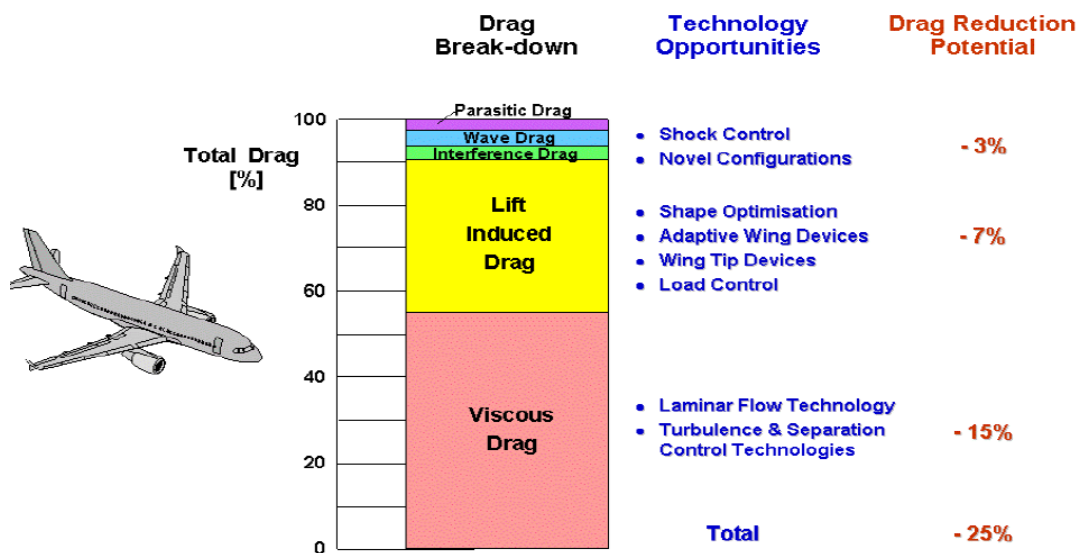
Modern air vehicle design has been increasingly driven by environmental as well as operational constraints. The growing impact of ecological and economical issues related to a change in climate has made urgent regulations and industrial actions for a future greening of air traffic. Among the most recognized within academic and industry community, the Advisory Council for Aeronautics Research in Europe (ACARE) 2020 goals (see ref. (2)) ask to halve the emitted carbon dioxides (CO<sub>2</sub>) and aircraft community noise as well as reducing emissions of nitrogen oxides (NO<sub>x</sub>) by 80%, based on the technological level of 2000. Therefore, environmental concerns, including emissions and noise, are gaining increasing importance in the design and operations of commercial aircraft. Taking into account the current prognoses for the growth in air traffic, the above mentioned challenges become even more significant. Despite the current dip in air traffic growth both Airbus and Boeing agree in their current market forecast that an annual growth of 5% for passenger services as well as for the air cargo market is likely. From the point of view of the aircraft design task, environmental objectives and constraints are translated into two main design targets: increase the aerodynamic efficiency and reduce the aircraft noise. To reach these objectives and to obtain substantial gain with respect to actual technology, new innovative and aggressive configurations have to be studied. Indeed, the environmental impact of air traffic is strongly affected by propulsion technology (fan, turbine, combustion, materials), air-frame technology (aerodynamics, structures, systems), air traffic management, physics

## 1. INTRODUCTION

---

of the atmosphere. With particular reference to the airframe impact, aerodynamics can play a significant role in fuel burn reduction. Figure 1.1 shows the relative importance of each aerodynamic drag contribution and the corresponding potential of reduction by investigating *ad-hoc* technologies. Here, the aerodynamic designer has a crucial task as he can provide several contributions:

- passively enhance the laminar flow, i.e. shaping the aircraft components to maximize the portion of laminar flow to reduce the viscous drag;
- reduce the lift-induced drag by investigating new wing configurations (e.g., high aspect ratio wings), optimizing the wing load and conceiving advanced wing tip devices;
- control the wave drag by designing shock-less wing airfoils;
- mitigate the aerodynamic impact of engine-nacelle-pylon interference by properly shaping the single components in order to avoid strong pressure jumps which would cause shock waves and flow separations.
- search for novel aircraft configurations to reduce the tailplane/interference drag.



**Figure 1.1: Aerodynamic drag reduction technology** - The figure shows the aerodynamic drag break-down and, for each contribution, the potential solutions to mitigate it (picture taken from ref. (3)).

As usual, in order to correctly define the design paths, one should look to the market. The regional market is already a large part of Air Transport System and in 2020 the share is estimated to rise to around 50% (according to references (4) and (5)). Therefore regional aircraft are contributors to pollution mainly around regional airports (noise, CO<sub>2</sub> and NO<sub>x</sub>). Both Airbus and Boeing agree in their forecasts that the market share of the regional and the single-aisle aircraft will be by far the biggest. In these market segments the typical mission flown is often not longer than 500 to 750 NM. Due to the shortness and high number of the mission cycles, an improvement in efficiency during cruise is not sufficient. Hence, the efficiency during all flight phases has to be improved. This means that the design optimization problem should be cast into a multi-(design)point and multi-objective form in order to take into account the aircraft performances with changing flight conditions along the whole mission profile.

Another important goal addressed by ACARE 2020 view is related to the improvement of the cost efficiency through the enhancement of the design workflow. The main objectives are the time reduction of the product development and the halving of the time-to-market. In this perspective, the present research aims at making a step towards bridging the gap between design stages (e.g. as conceived by Raymer (6) and summarized in table 1.1) through the coupling and exploitation of advanced analysis methods, reduced order/meta-modeling, optimization techniques and CAD-based tools towards the aerodynamic design of innovative aircraft configurations with reasonable computational resources. The introduction of physics-based surrogate models will allow to correctly drive the design process since the very early stages and, hence, to refine the evaluation of potentially cost/environment saving concepts.

## 1.2 Surrogate-Based Optimization

The intrinsic design complexity of modern aircraft relies more and more on the development and assessment of new theoretical methodologies capable of reducing or even replacing the experimental load. Moreover, theoretical methods are often used to further explore the trade-offs and alternatives when a decision about design path to be undertaken must be faced or a down selection among design candidates must be done. Two main characteristics are usually required to theoretical predictions in aircraft design and analysis: high-fidelity and low cost. High-fidelity is related to the capability

## 1. INTRODUCTION

---

of the theoretical method to reproduce “real-life” phenomena with a significant degree of accuracy (e.g., flow transition and separation, aerodynamic stall prediction). Indeed, due to global competition which pushes towards ever increasing technical and commercial requirements, the theoretical/numerical representation of analysis details and physical behaviour are applied in the early stages of the design process where the design space has few limitations and therefore the number of degrees of freedom is still large. Semi-empirical tools and rules, derived from classical configurations data, have been traditionally applied thanks to their computational efficiency. However they exhibit validity and flexibility issues in modern design as they loose accuracy when the design path moves away from conventional. Moreover, to reach the environmental objective of ACARE 2020, it is necessary to address non conventional configuration for which existing semi-empirical simulation tools are not applicable. Therefore, a strong effort has been done in the recent past to introduce potentially highly accurate design analysis methods both in geometry and physics modelling. The main drawback is that they are computationally expensive. For example, the solution of non-linear steady or unsteady aerodynamic flows by numerically solving the Navier-Stokes equations implies an amount of data storage, data handling and processor costs that may result very intensive even when implemented on modern state-of-art computing platforms. This turns out to be an even bigger issue when used within parametric studies, automated search or optimization loops which typically may require thousands analysis evaluations.

In order to speed up the analysis process while keeping a high level of fidelity, researchers from across the world are increasingly focusing on surrogate methodologies like meta-models or reduced order models. They can provide a compact, accurate and computationally efficient representation of the aircraft design performance index, usually referred to as objective or fitness function in an optimization context. Among the various surrogate models, here with the term *data-fits* or *meta-models* we will address all the methodologies which are based on the interpolation or reconstruction through response surfaces of the objective function. On the other hand, *reduced order models (ROM)* are mathematical models of the physical system under analysis (i.e., not just the objective function is considered) which contain fewer degrees of freedom than the original one and therefore it is relatively inexpensive to compute. Their use is primarily motivated by the need to have detailed knowledge of the physics together with an

## 1.2 Surrogate-Based Optimization

Conceptual design	Preliminary design	Detail design
Definition of requirements	Quick trade-offs and performance evaluation	Design the actual pieces to be built
Sketch of the general layout	Configuration design	Finalize weight and performance estimates
Identification of trade-offs Gross estimation of weight and cost	"Low-fidelity" tools	"High-fidelity" tools

**Table 1.1:** The aircraft design phases according to Raymer

efficient and reliable prediction tool. Both data-fits and reduced order models are very sensible to the so-called *training* phase, which consists in feeding the model with high-fidelity data and finding the set of parameters which best fit the model to the available data. For this purpose, effective sampling of the multi-dimensional design space is usually achieved by *Design of Experiments (DOE)* methods. The selected sampling points are evaluated with the high-fidelity tools and, depending on the adopted surrogate technique, design objectives and constraints or vector/scalar fields of interests are used to train the surrogate model. However, as we will discuss in the final chapters, sampling the design space for surrogate building purposes is not a trivial operation, as a trade-off exists between the exploration of the design solutions and the improvement of the surrogate accuracy near predicted minima. Indeed, we will show how a proper balance between these two concepts should be sought, especially when a shape optimization problem, implying multi-modal objective functions, constraints and highly non-linear characteristics, has to be faced.

### 1.2.1 The Aircraft Design Optimization Problem

A broad class of aircraft design applications can be numerically modeled with the minimization of a function  $f$  which depends on two sets of variables: the design variables  $\mathbf{x}$ , which the designer can directly control, and the state variables  $\mathbf{y}$  which provide the evolution of the system representing the underlying physics. The design problem can be formulated as a nonlinear programming problem:

## 1. INTRODUCTION

---

$$\begin{aligned}
& \min_{\mathbf{x}, \mathbf{y}} && f(\mathbf{x}, \mathbf{y}) \\
& \text{subject to} && r(\mathbf{x}, \mathbf{y}) = 0 \\
& && \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq 0, \\
& && \mathbf{x}_L \leq \mathbf{x} \leq \mathbf{x}_U
\end{aligned} \tag{1.1}$$

$f$  is the objective function which the designer wants to minimize to improve the performances. In aircraft design, typical objective functions are the weight, the noise, the drag, the aerodynamic efficiency or a combination of them.  $r(\mathbf{x}, \mathbf{y})$  is the state equations set which links the design variables and the state variables and it usually represents the governing laws, inspired by the physics, that the physical system must satisfy. In aerodynamic design, the state equations are modelled through computational fluid dynamics, e.g. the Navier Stokes equations, which relates scalar or vector field (state) variables, like pressure or velocity, to the aircraft component shape. This is made dependent on the design vector by means of a parameterization approach, which is another fundamental ingredient in a shape optimization problem. The vector  $\mathbf{g}(\mathbf{x}, \mathbf{y})$  and  $\mathbf{h}(\mathbf{x}, \mathbf{y})$  are filled respectively with inequality and equality constraint functions which must be satisfied to consider a design candidate feasible. Examples in aircraft design are the generation of a minimum lift level to balance the weight or a threshold pitching moment coefficient to allow for trim.  $\mathbf{x}_L$  and  $\mathbf{x}_U$  are the lower and upper bounds of the design variables and thus specify the range of allowable values for the design vector  $\mathbf{x}$ . The computational time required to solve this problem is basically affected by two parameters: given a vector  $\mathbf{x}^*$ , the cost for a single evaluations of  $f(\mathbf{x}^*, \mathbf{y})$  and  $\mathbf{g}, \mathbf{h}(\mathbf{x}^*, \mathbf{y})$ , which require the satisfaction of  $r(\mathbf{x}^*, \mathbf{y}) = 0$ , and the number of function evaluations required to effectively minimize the objective function. By adopting a surrogate model, we overcome the first of the two time-consuming factor. A surrogate model consists in replacing the expensive objective  $f$  and constraint functions  $\mathbf{g}, \mathbf{h}$  with less expensive, lower-fidelity models  $\hat{f}$  and  $\hat{\mathbf{g}}, \hat{\mathbf{h}}$ . In the next section, a brief overview of typical surrogate models will be given. Concerning the reduced order modelling, it can be observed that the dimensionality of the optimization problem is twofold: the state vector and design vector dimension. As the first one is usually much bigger than the second, model reduction can be applied to explicit the dependency of  $\mathbf{y}$  on  $\mathbf{x}$  and solve the state variables as functions of the design ones. In other words,



the reduced order optimization problem 1.1 can be cast in a simplified form as:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{subject to} \quad & \mathbf{c}(\mathbf{x}, \mathbf{y}) \leq 0 \end{aligned} \tag{1.2}$$

where the dependence on the state variables has been dropped and all the constraint functions have been enclosed in a single vector.

### 1.2.1.1 The importance of geometry parameterization

An important issue is the link between the response function and the design parameters. A proper parameterization has to be carefully chosen when using a surrogate model because the training phase has to include not only as much physics as possible (i.e., right selection of the high-fidelity solver) but also as much information about the design sensitivities as possible. As an example, let's consider the optimization of a wing airfoil for maximum natural laminar flow in transonic conditions. It is well known that one of the key design parameter is the leading edge radius: indeed, the front part of the airfoil should be shaped in order to avoid leading edge suction peak that would enhance the transition to turbulence and destroy the laminarity of the boundary layer. Even using an aerodynamic method able to catch the physics behind the laminar to turbulent transition, the design would fail if the chosen parameterization would not take into account the complete shape modification around the leading edge. Indeed, when representing the geometry of an aircraft or an aircraft component in any particular aerodynamic design optimization process, the choice of a proper mathematical representation can significantly alter the computational resources of the overall optimization process, the size and characteristics of the design space and the smoothness/feasibility of a design solution. Moreover, the coupling between the optimization and the parameterization methods should be carefully evaluated in order to avoid incompatibility issues. A classic example is the use of discrete coordinates as design variables in an evolutionary optimization process, where the resulting design space could be heavily populated with irregular, bumpy and not smooth geometries; thus, finding a realistic smooth optimum may be practically impossible. The geometry representation method also affects whether a meaningful optimum is contained in the design space and if an optimum design exists, whether or not it can be found. So, high-fidelity is not enough: geometry parameterization is another key point towards the optimal solution of a given aircraft design problem.

## 1. INTRODUCTION

---

### 1.2.2 Surrogate models

Surrogate models or meta-models have been introduced into the design optimization community to capture the salient features of an expensive high-fidelity model at low computational cost. In the aircraft design common practice, they are used as response functions in parametric analyses over the design space to understand the variations of the objective functions, as quick evaluator in an optimization problem or even in uncertainty quantification studies. Basically, surrogate models can be divided in three groups: data fits surrogates, multi-fidelity models and reduced order models.

#### 1.2.2.1 Data-fit models

Data fitting methods involve construction of an approximation or surrogate model using data (response values, gradients, and Hessians) generated from the original truth model. Global methods, often referred to as *response surface methods*, involve many points spread over the parameter ranges of interest. These surface fitting methods work in conjunction with the sampling methods and design of experiments methods. Some examples of global methods are:

**Polynomial Regression:**  $n^{th}$  order polynomial approximations computed using linear least squares regression methods;

**Kriging interpolation and Gaussian process:** it is a regression method used in geostatistical sciences which interpolates the value of a random field at an unobserved location from observations of its value at nearby locations. The algorithm used in the kriging process generates a  $C^2$ -continuous surface that exactly interpolates the data values. Similar to Kriging, a Gaussian process is a spatial interpolation method that assumes the outputs of the simulation model follow a multivariate normal distribution. The hyper-parameters governing the covariance matrix are obtained through Maximum Likelihood Estimation (MLE);

**Artificial Neural Networks:** a neural network consists of an interconnected group of artificial neurons. In most cases a neural network is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase;

**Radial Basis Functions:** Radial basis functions (RBFs) are functions whose value typically depends on the distance from a center point, called the centroid. The surrogate

model approximation is constructed as the weighted sum of individual radial basis functions;

**Moving Least Squares:** it can be considered a more specialized version of linear regression models. It provides a weighted least squares approach where the weighting is “moved” or recalculated for every new point where a prediction is desired;

### 1.2.2.2 Multi-fidelity models

A second type of surrogate is the hierarchical one (also called multi-fidelity or variable fidelity). In this case, the surrogate model is built with a lower fidelity approximation which, however, is still inspired by the physical behaviour of the system. Multi-fidelity models are classified according to the way they operate the fidelity reduction: examples in aerodynamics are coarser mesh discretization, same high-fidelity model but at higher residual convergence levels, simplified methods obtained by neglecting part of the basic physics of the high-fidelity (e.g., neglecting the effects of fluid viscosity and heat transfer into Navier-Stokes model allows to derive the Euler model).

### 1.2.2.3 Reduced order modelling

A reduced-order model (ROM) is mathematically derived from a high-fidelity model using a projection technique. It consists in computing a set of basis functions (e.g., eigenmodes, left singular vectors) from an ensemble of representative dataset of real scalar or vector fields, identifying how many of them capture the principal dynamics of the system and projecting the high-order system onto the retained basis. Hence, reduced order methods require the *a priori*, off-line solution of high-fidelity, very expensive governing equations in order to build the ensemble dataset. The advantage of reduced order models with respect to data fits is that they are derived by projection of the high-fidelity field solution rather than by interpolation of some quantities, thus having the potential to keep more physics within the approximation. The Proper Orthogonal Decomposition (POD) or Principal Component Analysis (PCA) is an elegant and powerful data-reduction method for non-linear physical systems. Its application to the aerodynamic optimization of aircraft components is the core of the present research work. The basic theory and techniques will be described later on in dedicated chapters.

## 1. INTRODUCTION

---

x	y	z
0.429604	0.756743	0.533522
0.034125	0.085996	0.216432
0.538876	0.463501	0.680638
0.213985	0.308366	0.734739
0.855647	0.022770	0.037772
0.195779	0.849284	0.082571
0.213159	0.445299	0.472457
0.317270	0.193087	0.123772
0.280803	0.298640	0.481686
0.249933	0.608013	0.963503

**Table 1.2:** Three-dimensional data set

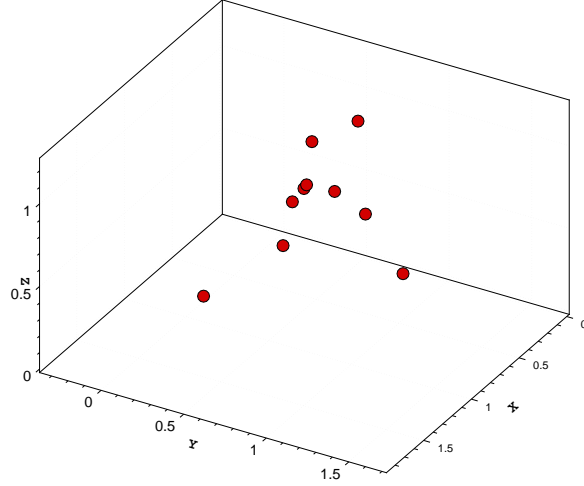
### 1.2.2.4 An example of dimensionality reduction

In this section, a very simple example of Proper Orthogonal Decomposition data reduction of a discrete set is presented to give a basic understanding of the technique. Consider a set of  $N = 10$  three-dimensional data ( $d = 3$ ), reported in table 1.2 and depicted in figure 1.2. These points can be projected from a three-dimensional space to smaller subspaces, e.g. two-dimensional ( $d = 2$ ) or one-dimensional ( $d = 1$ ). Here, a tutorial on how to perform such a projection in an “optimal” way with POD is illustrated. First of all, let’s define the mean, variance and covariance of the data set:

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i; \quad \sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2; \quad C_{xy} = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})$$

It the variance is a measure of how far a set of numbers are spread out from each other, the covariance is a measure of how two variables change together.

Variance and covariance can be summarized in the so-called covariance matrix, where the  $(i^{th}, j^{th})$  element is the covariance between the variable  $i$  and  $j$ . Obviously, the elements along the main diagonal represent the variance and the matrix is symmetric positive semi-definite. In the present case, the mean values are  $\bar{x} = 0.332918$ ,  $\bar{y} = 0.403169$ ,  $\bar{z} = 0.432709$  while the covariance matrix has the following form:



**Figure 1.2:** Three-dimensional data set, graphical representation

$$C = \begin{bmatrix} 0.052381 & -0.013409 & -0.012827 \\ -0.013409 & 0.075640 & 0.028309 \\ -0.012827 & 0.028309 & 0.096596 \end{bmatrix}$$

The variance along the  $z$ -direction is almost double than along  $x$  and this clearly indicates that, being the variance of the data set dependent on the projection direction, there should exist a peculiar unit vector (data reduction to  $d = 1$ ) which maximizes the variance when projecting the three-dimensional data onto it. Similarly, there should exist a plane which maximizes the covariance when projecting the data set onto it and minimizes the variance along the normal direction. These “optimal” directions can be determined by finding the eigenvalues of the covariance matrix and their corresponding eigenvectors. The eigenvalues and eigenvectors are:

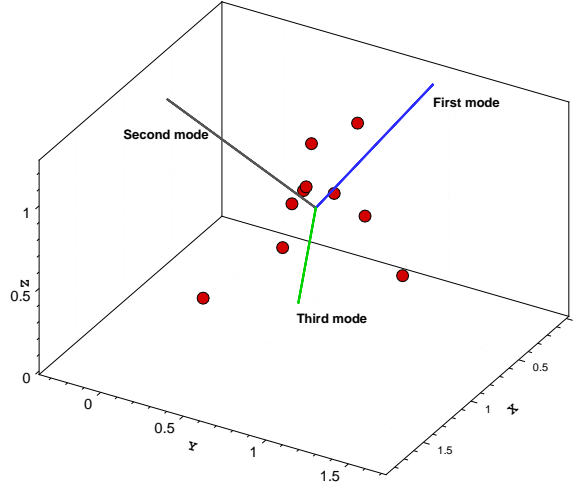
$$\lambda_1 = 0.046261; \quad \lambda_2 = 0.057225; \quad \lambda_3 = 0.121131;$$

$$[\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3] = \begin{bmatrix} 0.9083706 & -0.3302926 & -0.2564559 \\ 0.4181497 & 0.7120389 & 0.5640491 \\ -0.0036946 & -0.6196026 & 0.7849070 \end{bmatrix}$$

In figure 1.3 the three eigenvectors, centered on the ensemble mean value, are plotted on the same graph with data points (green, grey and blue colors respectively for  $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ ): not only they are perpendicular, but also the eigenvector with the corresponding bigger eigenvalue ( $\mathbf{v}_1$ , blue vector in the figure) provides the best linear

## 1. INTRODUCTION

---



**Figure 1.3:** POD basis graphical representation

fit of the data. This is because the magnitude of an eigenvalue gives the variance of the dataset when projecting along the corresponding eigenvector. Hence, ranking the eigenvalues means to introduce an ordering of the eigenvectors according to their “importance” in fitting the original data. By ignoring the last eigenvectors, as they are the less significant, the original data set will be described in terms of a reduced set of dimensions. This is the main concept behind the notion of dimensionality reduction. Now, suppose that we keep just two out of three eigenvectors (known also as POD modes): in this case, a *two*-dimensional data set is derived from the original *three*-dimensional one by multiplying the centered data (obtained by subtracting the mean value to the initial ensemble) with  $\mathbf{v}_2, \mathbf{v}_3$ :

$$\begin{bmatrix} 0.096686 & 0.353573 & 0.100812 \\ -0.298793 & -0.317174 & -0.216277 \\ 0.205958 & 0.060331 & 0.247929 \\ -0.118933 & -0.094804 & 0.302030 \\ 0.522729 & -0.380400 & -0.394937 \\ -0.137140 & 0.446114 & -0.350138 \\ -0.119759 & 0.042129 & 0.039748 \\ -0.015648 & -0.210083 & -0.308937 \\ -0.052115 & -0.104529 & 0.048977 \\ -0.082985 & 0.204843 & 0.530793 \end{bmatrix} \begin{bmatrix} -0.3302926 & -0.2564559 \\ 0.7120389 & 0.5640491 \\ -0.6196026 & 0.7849070 \end{bmatrix} = \begin{bmatrix} 0.1573598 & 0.2537654 \\ 0.0068553 & -0.2720319 \\ -0.1786861 & 0.1758118 \\ -0.2153595 & 0.2140922 \\ -0.1988088 & -0.6586102 \\ 0.5798934 & 0.0119745 \\ 0.0449246 & 0.0856744 \\ 0.0469992 & -0.3569709 \\ -0.0875623 & -0.0071520 \\ -0.1556155 & 0.5534469 \end{bmatrix}$$

The obtained data, known as POD coefficients, represent the projections of the centered original data onto the two directions with the most significant variance. Hence, a new three-dimensional data set can be reconstructed by multiplying the POD coefficients with  $\mathbf{v}_2, \mathbf{v}_3$ :

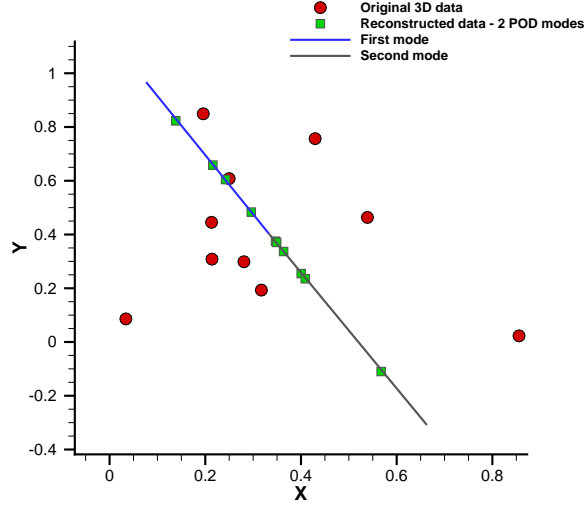
$$\begin{bmatrix} 0.1573598 & 0.2537654 \\ 0.0068553 & -0.2720319 \\ -0.1786861 & 0.1758118 \\ -0.2153595 & 0.2140922 \\ -0.1988088 & -0.6586102 \\ 0.5798934 & 0.0119745 \\ 0.0449246 & 0.0856744 \\ 0.0469992 & -0.3569709 \\ -0.0875623 & -0.0071520 \\ -0.1556155 & 0.5534469 \end{bmatrix} \begin{bmatrix} -0.33029 & 0.71204 & -0.61960 \\ -0.25646 & 0.56405 & 0.78491 \end{bmatrix} = \begin{bmatrix} -0.117054 & 0.255182 & 0.101682 \\ 0.067500 & -0.148558 & -0.217767 \\ 0.013931 & -0.028065 & 0.248710 \\ 0.016226 & -0.032586 & 0.301480 \\ 0.234570 & -0.513048 & -0.393765 \\ -0.194605 & 0.419661 & -0.349905 \\ -0.036810 & 0.080313 & 0.039411 \\ 0.076024 & -0.167884 & -0.309310 \\ 0.030755 & -0.066382 & 0.048640 \\ -0.090536 & 0.201367 & 0.530824 \end{bmatrix}$$

Figure 1.4 plots the reconstructed three-dimensional data compared to the original ones on the XY plane. Such a representation is very useful because the less energetic mode, i.e. the first eigenvector, has almost a zero-valued component along the z-axis, meaning that it lies in a plane perpendicular to the z-axis: because of their mutual orthogonality, the first two modes necessarily define a plane almost parallel to the z-axis along which the reconstructed data are placed as only two eigenvectors have been used. Hence, a visualization on the XY plane is particularly interesting to understand the result of the projection. It is evident that the reconstruction will lose some information, but nevertheless a good approximation of the original ensemble is obtained. The approximated data can be represented in three dimensions, but they are basically two-dimensional as they can be represented by two basis vector only. Therefore, the original three-dimensional set of data was reduced to a two-dimensional set of data represented by the coordinates in the  $\mathbf{v}_2, \mathbf{v}_3$  plane.

It must be underlined that in the proposed example the data ensemble has been generated randomly, so as no clear patterns or coherent trends can be identified between the lines. Indeed, when applying the POD technique to real cases, where the ensemble is assembled on results coming from a physics-based model, the main challenge will be to extract the most significant features through the model reduction. Moreover, when the POD concept is extended to the handling of large data sets as in computational fluid dynamics (CFD), a bigger potential can be seen to reduce such multi-dimensional

## 1. INTRODUCTION

---



**Figure 1.4:** 3D data reconstruction with 2 POD modes - XY plane projection

problems. In particular, a significant saving in computational time, resources and storing is expected to be achieved.

### 1.3 Aim and objectives

The main objective of the present research is to investigate a surrogate-based optimization method for aerodynamic design which adopts POD-based techniques as approximation models for computational efficiency and improved accuracy. Two-dimensional Reynolds-averaged Navier-Stokes flows are considered, but the proposed methodologies, dealing with linear algebra techniques, can be easily extended to three-dimensional flows. The suitability of the model reduction is particularly studied for the treatment of transonic conditions. To this aims, several contributions and steps have been identified and accomplished.

- Build a mathematical model for dimensionality reduction through Proper Orthogonal Decomposition;
- Develop a physics-based global surrogate model by coupling the POD module with a data-fit interpolation method. This step is needed to make the meta-model able to predict the flow solution outside the training ensemble;



- Develop an alternative computational approach to overcome the limits of surrogate models in predicting transonic flows. In particular, a zonal approach can be effective as it decomposes the flow domain in two parts: a full-order domain where the high-fidelity analysis is carried out and a reduced order domain where the POD prediction is applied;
- Develop adaptive design of experiments techniques to cleverly train the POD module when highly non-linear, rapidly varying design landscapes and large design spaces have to be explored;
- Validate the POD-based global surrogate in a two-dimensional transonic case;
- Integrate the POD surrogate model into an evolutionary optimization workflow;
- Perform a surrogate-based evolutionary optimization of a two-dimensional transonic case by exploiting the validated approach and comparing it with standard techniques to understand *pros* and *cons* of the physics-based meta-model;

## 1.4 Thesis outline

The thesis can be ideally divided into three main parts. The first is devoted to the illustration of state-of-the-art computational tools and computational procedures for aerodynamic design. The second is devoted to the presentation of POD theory and POD-based reduced order models. The third part contains the application of surrogate-based optimizations to aerodynamic design in transonic flow. The next chapter is dedicated to an extensive literature review about surrogate-based optimization and POD-based modelling. In chapter three, an overview of the ingredients for aerodynamic optimization are presented. An advanced geometry parameterization with CAD integration is proposed as an accurate method to describe aerodynamic shapes and to be consistently used in a design optimization process. Therefore, the governing equations of Fluid Dynamics are introduced, the aerodynamic flow solver ZEN is described as it has been used to generate the surrogate training solutions and it provides the reference aerodynamic evaluator in our optimization chain. The evolutionary optimization tool is briefly described and an example of aircraft wing-body aerodynamic optimization in

## 1. INTRODUCTION

---

transonic conditions is shown to provide a state-of-the-art approach and further motivations to the present research. Chapter four is devoted to the POD surrogate theory with special focus on the singular value decomposition solution and pseudo-continuous representation of POD coefficients. Chapter five proposes some techniques to enhance the reduced order prediction in presence of a highly non-linear design optimization. Moreover, adaptive design of experiments techniques based on the improvement of POD modal coefficients and basis vectors are discussed. Finally, chapter six focuses on the application of adaptive and non-adaptive, data-fit surrogates and POD-based reduced order models in aerodynamic optimization: the optimization a two-dimensional airfoil in transonic flow with and without surrogates is presented and a detailed analysis of the obtained results lay the foundations for further work and improvements.

## 2

# Literature review

This chapter proposes a survey of the most relevant literature articles concerning surrogate-based evolutionary optimization and reduced order modelling with Proper Orthogonal Decomposition applied to aircraft aerodynamic design. The topics have been widely discussed in the recent past, thanks to their innovative character and broad application areas. The introduction of surrogate models as fitness approximation within an evolutionary optimization system mitigates the demand for large computational resources associated to such search algorithms, allowing to find a proper balance between the complete exploration of huge design spaces and limited cost. To this aim, reduced order modelling through POD is a step forward, as a modal decomposition of an ensemble of functions, derived from numerical simulations, is performed to extract the most relevant patterns in the data set. Hence, compared to standard, interpolating meta-models which are usually trained on an integral function representing the objective, reduced order models should assure a deeper insight into the modelled phenomena.

## 2.1 Surrogate-based Optimization

Surrogate-based optimization (SBO) has been introduced to tackle the number of function evaluations in many engineering optimization problems. This represents a special challenge in the field of global optimization as state-of-the-art methods often requires more function evaluations than can be comfortably affordable. A well-established approach consists in fitting some kind of response functions to basic data obtained by evaluating the objectives and constraints at a few points. The resulting surfaces, af-

## 2. LITERATURE REVIEW

---

fordable at low cost, can provide fast answers in terms of trade-off analysis and optimization as well as just an intuitive sketch behaviour by means of simple visualization. The basic process is the following:

1. design space sampling: once the design variables have been chosen, a sampling plan is defined and some initial sample designs are analysed with an accurate solver;
2. surrogate model selection and construction: a surrogate model type is selected and used to build a meta-model of the underlying problem;
3. model validation: the model is checked according to some statistical metrics and, if not enough accurate, a search is carried out using the model to identify new design points for analysis;
4. model updating: the new results are added to those already available and a new meta-model is built (repeating the last three steps);
5. optimization: the refined surrogate is used to provide objective/constraint functions;

As SBO covers so many topics, the literature on the subject is huge. A plenty of ideas have been proposed in the last twenty years, classified for design space dimensions, surrogate methods, search algorithms, updating algorithms, application areas. Hence, an exhaustive survey of all the possible ideas for each topic and all the possible combination of them would go beyond the scope of the present research. Here we take a more in depth look at the various methods of constructing a surrogate model and, in particular, at optimization assisted with the surrogate. Jones et al. (7), among the first, proposed a response surface methodology based on modelling the objective and constraint functions with stochastic processes (Kriging). The so-called Design and Analysis of Computed Experiments (DACE) stochastic process model was built as a sum of regression terms and normally distributed error terms. The main conceptual assumption was that the lack of fit due only to the regression terms can be considered as entirely due to modelling error, not measurement error or noise, because the training data are derived from a deterministic simulation. Hence, by assuming that

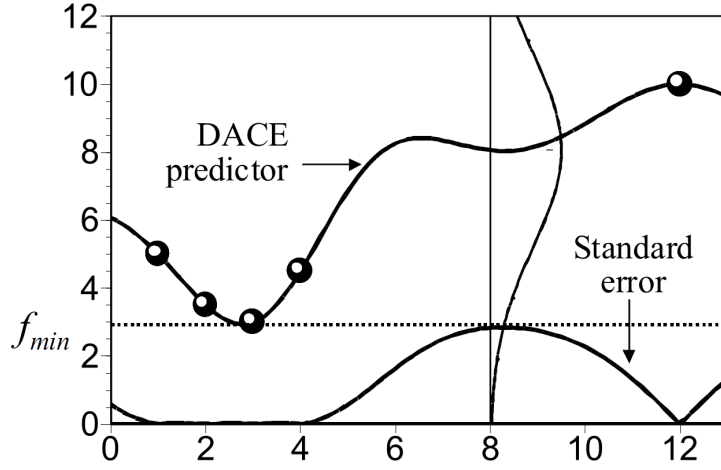
the errors at different points in the design space are not independent and the correlation between them is related to the distance between the computed points, the authors came up with an interpolating surrogate model able to provide not only the prediction of objectives/constraints at a desired sample point, but also an estimation of the approximation error. After the construction of such a surrogate model, this last powerful property is exploited to build an Efficient Global Optimization (EGO), which can be considered as the progenitor of a long and still in development chain of SBO methods. Indeed, they found a proper balancing between the need to exploit the approximation surface (by sampling where it is minimized) with the need to improve the approximation (by sampling where prediction error may be high). This was done by introducing the Expected Improvement (EI) concept, already proposed by Schonlau (8), that is an auxiliary function to be maximized instead of the original objective. Figure 2.1 well explains the underlying idea. At each point along the x-axis, a normal density function with the mean and standard deviation suggested by the DACE predictor can be drawn. The tail of such distributions extends below the current best function value  $f_{min}$ . The integral of the area below  $f_{min}$  represents the probability of improvement. By weighting the possible improvements by the associated density value, the Expected Improvement is obtained. Figure 2.2 shows an example taken from (7). The red curves represent the DACE prediction built on blue sample points, while the green curve shows the Expected Improvement curves. It is easy to see that sampling at a point where this auxiliary function is maximized improves both the local (exploitation) and global (exploration) search.

In a further work, Jones (1) proposed a taxonomy of global SBO methods. Seven methods were identified and classified on whether they were interpolating (cubic splines, thin-plate splines, multiquadrics, kriging) or not (quadratic polynomials), whether they provided statistical information (kriging) or not (splines) and whether the method for selecting search points (updating the model by adding new sample points) was two-stage (probability/expected improvement) or one-stage (goal-seeking, credibility function). Figure 2.3 reports the considered taxonomy. The seven SBO methods described hereinafter refer to those described in the figure.

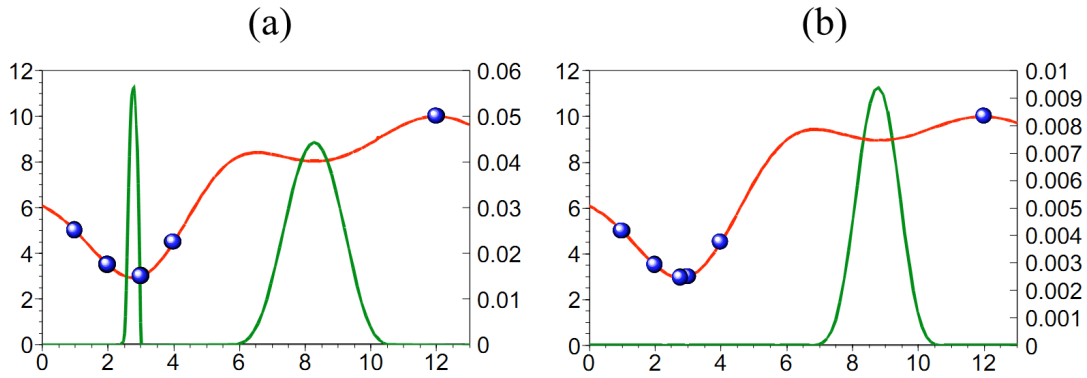
The first two methods consisted in four steps: fitting a surface through sample points, finding the minimum of the surface, evaluating the function at the surface minimum and then iterating. In particular, the first one, based on a quadratic polynomial

## 2. LITERATURE REVIEW

---



**Figure 2.1: DACE prediction error** - The figure shows how a normal density function can be built at each point having the standard deviation equal to the DACE prediction error.



**Figure 2.2: Expected Improvement measure to improve the goodness-of-fit** - (a) The expected improvement function when only five points have been sampled; (b) the expected improvement function after adding a point at  $x=2.8$ . In both (a) and (b) the left scale is for the objective function and the right scale is for the expected improvement

Kind of Response Surface			Method for selecting search points				
			Two-stage approach: first fit a surface, then find the next iterate by optimizing an auxiliary function based on the surface				One stage approach: evaluate hypotheses about optimum based on implications for the response surface
			Minimize the Response Surface	Minimize a Lower Bounding Function	Maximize the Probability of Improvement	Maximize Expected Improvement	Goal seeking: find point that achieves a given target
Not interpolating (smoothing)	Quadratic polynomials and other regression models		1				
Interpolating	Fixed basis functions. NO statistics.	Thin-plate splines, Hardy multiquadrics	2				6
	Tuned basis functions. Statistical interpretation	Kriging	3	4	5		7

Figure 2.3: Taxonomy of response surface-based global optimization methods, according to Jones (1) -

surface, completely failed due to a lack of fitting. Even the second method, built on kriging or spline interpolating surfaces, easily missed the global optimum, but at least converged to a local minimum. Torn and Zilinskas (9) proved that, in order to converge to the global optimum for a general continuous function, the sequence of iterates must be dense. In other words, we are asking the method to converge to every point in the domain. The practical lesson of the theorem is that any globally convergent method must have a feature that forces it to pay attention to parts of the space that have been relatively unexplored and, from time to time, to go back and sample in these regions. Methods 1 and 2 failed to find a global minimum in our examples because they have no such feature. As discussed by Alexandrov et. al. (10), a comfortable way to improve such methods and to ensure convergence to a critical point is to force the gradient of the surface to match the gradient of the function whenever the search stagnates. Alexandrov showed that this additional condition is not sufficient: by using also a trust region approach, a locally convergent method was developed. In this context, it is worth to mention the work from Booker et.al. (11), who showed how response surfaces can be used to accelerate a derivative-free method of local optimization.

Proceeding with the proposed taxonomy, Jones studied three more methods based on kriging ability to estimate potential error in its predictions. The first one (Method

## 2. LITERATURE REVIEW

---

3) minimizes an auxiliary function called statistical lower bounding function obtained by subtracting several standard errors from the predictor. However, the successive iterates resulted to not be dense, hence the search could fail to find the global minimum according to Torn and Zilinskas theorem. Method 4 built the next iterate by maximizing the estimated probability that the function value at a point would be better than a pre-defined target  $T$ . Obviously, the search would depend on the value of  $T$ , which we do not know a-priori, rather it is the goal of the search process. Anyway, by computing several search points for several values of  $T$ , an enhanced method was obtained which showed promising results. The last kriging-based Method 5 maximized the expected improvement, as described above. Methods 3–5, relying upon the standard error computed in kriging, could perform poorly if initial sample was highly deceptive. Deceptive samples could cause the kriging standard error to underestimate the true error in the predictor and, as a result, Methods 3–5 might converge prematurely or slowly. The use of several targets in Enhanced Method 4 seemed to reduce the negative impact of a deceptive initial sample, which is one reason why this approach was considered very promising. The five methods reviewed so far were “two-stage” methods, because the meta-model was first built on the sampling set and then exploited through auxiliary functions optimization to search for new updating points. In the same work, Jones turned his attention to “one-stage” methods with the aim of completely avoiding to be diverted by deceptive samples.

These methods make some hypotheses about the location of the optimum and then, by using the response surface mathematical support, measure the “credibility” of such hypotheses. In Method 6 Jones assumed that a known goal  $f^*$  value should be reached for the objective function. The next iterate was the point  $x^*$  where, based on some mathematical definitions, it was “most credible” to find the value  $f^*$ . This method was found to converge quickly to the goal. In Method 7, the author again aimed at find the global minimum of the objective function, but this time the minimal function value was not known neither fixed in advance. This case was handled by computing several search points using several values of  $f^*$ , just as several values of  $T$  were used in Enhanced Method 4. Gutmann (12) reported excellent numerical results for a spline-based implementation of Method 7 and proved the convergence of the method. Compared to previous methods, Method 7 required a high number of true function evaluations to find the global optimum, but, as Jones wrote, “this is the price we pay



for the additional robustness”. An overview of SBO techniques was presented also by Queipo et al. (13) and Simpson et al. (14). They covered some of the most popular methods in design space sampling, surrogate model construction, model selection and validation, sensitivity analysis, and surrogate-based optimization. Forrester and Keane (15) recently proposed a review of some advances in surrogate-based optimization. An important lesson learned is that only calling the true function can confirm the results coming from the surrogate model. Indeed, the path towards the global optimum is made of iterative steps where, even exploiting some surrogate model, only the best results coming from the true function evaluations are taken as optimal or sub-optimal design. The true function evaluation has to be also invoked to improve the surrogate model. With the term “in-fill criteria” it is usually meant some principles which allow to intelligently place new points (in-fill points) at which the true function should be called. The selection of infill points, also referred to as adaptive sampling or model updating, represent the core of a surrogate-based optimization method and helps to improve the surrogate prediction in promising areas of the objective space.

The right choice of the number of points which the initial sampling plan would comprise and the ratio between initial/in-fill points has been the focus of several recent studies. However, it must be underlined that no universal rules exist, as each choice should be carefully evaluated according to the design problem (e.g., number of variables, computational budget, type of surrogate). Forrester and Keane assumed that there is a maximum budget of function evaluations, so as to define the number of points as a fraction of this budget. They identified three main cases according to the aim of the surrogate construction: pure visualization and design space comprehension, model exploitation and balanced exploration/exploitation. In the first case, the sampling plan should contain all of budgeted points as no further refinement of the model is foreseen. In the exploitation case, the surrogate can be used as the basis for an in-fill criterion, that means some computational budget must be saved for adding points to improve the model. They also proposed to reserve less than one half points to the exploitation phase as a small amount of surrogate enhancement is possible during the in-fill process. In the third case, that is two-stage balanced exploitation/exploration in-fill criterion, as also shown by Sobester et al. (16), they suggested to employ one third of the points in the initial sample while saving the remaining for the in-fill stage. Indeed, such balanced methods rely less on the initial prediction and so fewer points are required. Concerning

## 2. LITERATURE REVIEW

the choice of the surrogate, the authors observed that it should depend on the problem size, i.e. the dimensionality of the design space, the expected complexity, the cost of the true analyses and the in-fill strategy to be adopted. They proposed their own SBO methods taxonomy, shown in table 2.4.

sample plan:infill points ratio				$\leq \infty$		$> 2 : 1$	$\approx 1 : 2$	$< 1 : 2$	
				comprehension		optimization			
				simple landscape	complex landscape	local search	$P[I(\mathbf{x})]$ , $E[I(\mathbf{x})]$	goal seeking	conditional lower bound
$k > 20$	$n > 500$	SVR		✓	✓	✓			
		fixed bases e.g. cubic, thin plate		✓		✓		✓	
polynomials		✓							
$k < 20$	$n < 500$	MLS, parametric bases	e.g. multi-quadric	✓	✓	✓		✓	
			Gaussian bases e.g. Kriging	✓	✓	✓	✓	✓	✓

**Figure 2.4: SBO methods taxonomy according to Forrester** - SBO methods are classified according to problem size  $k$ , computational budget in terms of number of true computations  $n$  and initial/in-fill samples ratio

However, for a given problem, there is not a general rule. The proper choice could come up past various model selection and validation criteria. The accuracy of a number of surrogates could be compared by assessing their ability to predict a validation data set. Therefore, part of the true computed data should be used for validation purposes only and not for model training. This approach can be infeasible when the true evaluations is computationally expensive. To overcome this issue, Goel et al. (17) proposed a weighted average of an ensemble of surrogates. For example, a better model can be achieved by combining Kriging, which might accurately predict the non-linear aspects of a function, and polynomials to better capture the regression trends. Forrester also underlined that some in-fill criteria and certain surrogate models are somewhat intimately connected. For a surrogate model to be considered suitable for a give in-fill criterion, the mathematical machinery of the surrogate should exhibit the capability to adapt to unexpected, local non-linear behaviour of the true function to be mimicked. From this point of view, polynomials can be immediately excluded since a very high order would be required to match this capability, implying a high number of sampling points. Figure 2.5, taken from Forrester research work, shows how the convergence to a local optimum can be achieved by simply minimizing the surrogate, evaluating the true

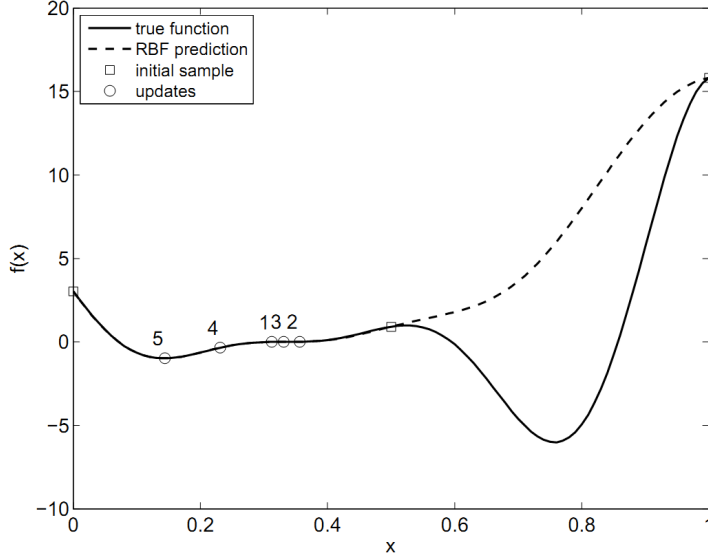
function at the minimum point and in-filling it for updating the model. In general, a global search would require a surrogate model able to provide an estimate of the error it commits when predicting. Thus, the authors suggested to use Gaussian process based methods like Kriging, although citing the work of Gutmann (12) as an example of one-stage goal seeking approach employing various radial basis functions. Finally, some interesting suitable convergence criterion to stop the surrogate in-fill process were proposed. In an exploitation case, i.e. when minimizing the surrogate prediction as in Figure 2.5, one can rather obviously choose to stop when no further significant improvement is detected. On the other hand, when an exploration method is employed, one is interested in obtaining a satisfying prediction everywhere, so that he can decide to stop the in-filling when some generalization error metrics, e.g. cross-validation, falls below a certain threshold. When using the probability or expectation of improvement, a natural choice is to consider the algorithm converged when the probability is very low or the expected improvement drops below a percentage of the range of observed objective function values. However, the authors also observed that discussing on convergence criterion may be interesting and fruitful, but "in many real engineering problems we actually stop when we run out of available time or resources, dictated by design cycle scheduling or costs". This is what typically happens in aerodynamic design, where the high-dimensionality of the design space and expensive computer simulations often do not allow to reach the global optimum of the design problem but suggest to consider even a premature, sub-optimal solution as a converged point.

## 2.2 Proper Orthogonal Decomposition

Proper Orthogonal Decomposition is exploited in the present research as a model reduction technique. A wide and comprehensive review of POD-based applications can be found in (18). Following the classification made by Antoulas et al. (19), they can be based on the singular value decomposition (SVD) and on moment matching. The Karhunen-Loève expansion turns out to be a popular method for both linear and non-linear systems and, hence, an appropriate choice for model order reduction in non-linear systems Euler or Navier-Stokes equations in aerodynamics. The technique was proposed by several authors at different times, in different fields and under a variety of names (20). It is essentially a linear transformation to diagonalize a given matrix. Due

## 2. LITERATURE REVIEW

---



**Figure 2.5: Example of SBO minimizing the predictor** - Starting from squares (initial samples), the method converges to a local minimum after adding five in-fill points (circles) in five steps of surrogate minimization

to the large number of computations required to extract the matrix eigenvectors, this technique was left aside until the increase of computational power produced efficient algorithms to get rid of this issue. The Proper Orthogonal Decomposition (also known as Principal Component Analysis) and the Singular Value Decomposition are generally treated as the same thing, however it must be underlined that the second technique is just a method of solution of the orthogonal basis, hence they are not strictly the same.

In the perspective of a fluid dynamic problem, the POD can be defined as a statistically-derived process which provides a mathematical representation of the high-energy components of a fluid flow field. This is done by decomposing the observed structure into a set of uncorrelated linear components which provide a low-dimensional representation of the problem. The components are the eigenfunctions of a correlation tensor and the expansion is optimal in the sense that the POD eigenfunctions maximize the total energy captured in each co-ordinate direction, subject to orthogonality constraints. Restricting the overview to the fluid dynamics applications, the method has been originally used in stochastic turbulence problems (21), where the POD eigenfunctions were related to the characteristic eddies of the turbulence field. The method has also been used in steady aerodynamic analysis such as the design of inviscid aerofoils

## 2.2 Proper Orthogonal Decomposition

---

by LeGresley and Alonso (22) and parametric studies by Epureanu et al. (23) and Bui-Thanh et al. (24, 25).

Concerning reduced order models derivation, Sirovich (26) introduced the method of snapshots as a way for efficiently determining the POD basis functions or modes for large problems. This technique can be used for a variety of applications, including derivation of reduced-order dynamical models for fluid dynamic applications as in Holmes et al. (27). Combined with CFD and unsteady aerodynamics, the method of snapshots has been widely used as in Dowell et al. (28) and Hall et al. (29). The general approach is to first compute a set of instantaneous flow solutions or snapshots and then apply the POD process to extract an optimal set of basis functions, where optimal means that the error between the originally computed and the reconstructed data is minimized. Once built the optimal orthogonal basis, reduced-order models can be derived by projecting the model onto the reduced space spanned by the POD modes. Therefore, the original problem, formulated in terms of non-linear partial differential equations as Navier Stokes model, can be converted into a small system of ordinary differential equations which can be solved efficiently.

Everson and Sirovich (30) have presented a variation of the basic POD method to handle incomplete data sets (also known as “gappy POD”). The method relies on a least square approximation, built on known data, to reconstruct an incomplete snapshot. Indeed, once computed the POD modes from the known data, an incomplete data vector can be reconstructed accurately by imposing the optimal conditions and solving the resulting linear system of equations. Another approach is also proposed, i.e. when the snapshots themselves are damaged or incomplete. In this case, an iterative method can be used to derive the POD basis, which is in turn used to reconstruct the incomplete data. This method has been successfully applied for the reconstruction of human face images, from partial data with 25% of the data missing.

The “gappy POD” method has been also exploited by Bui-Thanh et al. (24, 25) in transonic flow analysis and optimization. The case considered is a NACA 0012 aerofoil at a free-stream Mach number of 0.8. The POD bases were created from 51 snapshots. These were computed at uniformly spaced values of the angle of attack in the interval  $[-1.25^\circ, 1.25^\circ]$ . An incomplete flow field was generated by a computation of the flow solution at an angle of attack of  $0.77^\circ$  and retaining only 121 surface pressure values out of a total number of 6369 pressure values throughout the whole flow field. With just

## 2. LITERATURE REVIEW

---

this limited surface pressure data available, the complete pressure field was determined accurately with only six POD modes. Moreover, in this work the authors examined the sensitivity of the reconstruction result to both the quantity and location of this surface data. In order to select a limited number of pressure measurements, a heuristic approach for unsteady flows was adopted. This approach suggests that sensors should be located in areas of high modal activity. The POD modes of a flow often exhibit sinusoidal spatial variation and sensors placed at local POD modal minima and maxima yield effective flow sensing results. This heuristic procedure was applied in that work to study the sensitivity of the reconstruction results to the amount of available data. An initial configuration of 11 measurement points corresponding to spatial optima of the first POD mode plus a few other points near the leading edge, where all POD modes were seen to vary rapidly, was chosen. Additional measurements were then considered by adding in turn the spatial optima of modes 2, 3, 4, 5 and 6 resulting in studies with 15, 21, 29, 31 and 39 sensing points respectively. From this work it was observed that the percentage error between the exact and reconstructed pressure measurements was very low even with a very small number of sensors, confirming the effectiveness of the heuristic sensor placement algorithm. It was also noticed that subsequent reduction in the error diminishes as higher modes are considered in the sensing which is consistent with the fact that subsequent modes constitute progressively less of the total energy. The POD eigenvalues can therefore be used not only to select the number of modes but also to choose an appropriate number of sensors. A further investigation was conducted by creating a set of POD basis vectors from an incomplete set of snapshots. Again, the NACA 0012 aerofoil at a free-stream Mach number of 0.8 was considered. A 26 snapshot ensemble was used with steady pressure solutions at angles of attack in the interval  $[0^\circ, 1.25^\circ]$ . To create the incomplete snapshot set, 30% of the pressure data of each snapshot was discarded randomly. The data was repaired by first repairing the missing data points in each snapshot with the average over the available data at that point. Thus a new ensemble of data was created which had no missing values. With the new ensemble, a first approximation to the POD basis was constructed. Then each snapshot in the ensemble was repaired using the first approximation of the POD basis. The repaired ensemble was then used to construct a second approximation to the POD basis. An iterative procedure was adopted and stopped after 50 iterations. From the convergence of the POD eigenvalue spectrum of the incomplete ensemble it

## 2.2 Proper Orthogonal Decomposition

---

was observed that after one iteration, the first two eigenvalues converged. After 45 iterations, only the first five eigenvalues converged; however these accounted for almost all of the flow energy.

From technical literature analysis, the use of POD methods for capturing the time variation in unsteady fluid dynamics problems is widespread, while few papers focusing on its application to parametric variation or even shape modification problems can be found. In the same works by Bui-Thanh et al. (24, 25), the authors apply the POD technique to steady transonic external aerodynamic problem. In both works, all snapshots were computed by an inviscid steady-state CFD code which uses a finite volume formulation. In (24) the POD technique was coupled with a cubic-spline interpolation method in order to develop low-order models that capture the variation in parameters. The problem considered in this work is steady flow about the NACA 0012 aerofoil with varying angle of attack and Mach number. The Mach number range considered  $[0.75, 0.85]$  was divided into 20 uniform intervals and the angle of attack range  $[0, 1.25^\circ]$  was divided into 10 uniform intervals. This resulted in a total number of snapshots in the ensemble of 231. Based on this snapshot set, interpolation was used to predict the flow pressure contours at any Mach number and angle of attack within the range considered. When the pressure flow field was predicted at an angle of attack of  $0.45^\circ$  which was not one of the snapshots and a Mach number of 0.8, with twenty-five eigenfunctions the contours of the reduced order model matched closely with the computed one. When the pressure flow field was predicted at an angle of attack of  $0.5^\circ$  and a Mach number of 0.812 which was not one of the snapshots, thirty eigenfunctions were required to achieve the desired level of accuracy. This indicated that the prediction is more sensitive to Mach number rather than the angle of attack. When the pressure flow field was predicted at an angle of attack of  $0.45^\circ$  and a Mach number of 0.812, as none of these values were used to generate the snapshots, 40 POD modes offered a satisfactory level of accuracy. These results showed that the POD method combined with interpolation allows models to be derived that accurately predict steady-state pressure fields over a range of parameter values. However, it is emphasized that in order for the interpolated result to be reliable, the properties of interest must vary smoothly with the parameters under consideration. It has been stated that *the approach can be extended to the case where more than two parameters vary and which may include geometrical properties in order to apply the models in an optimization context*. The POD combined

## 2. LITERATURE REVIEW

---

with a response surface method was employed by Tang et al. (31) for the reconstruction and prediction of aerodynamic and aerothermal solutions of an X-34 configuration. It was reported that this module proved to be not only computationally more efficient than the low-level engineering methods, but also as accurate as the high-level CFD methods, making it valid for MDO and real-time applications.

In the field of aerodynamic shape optimization, the work by LeGresley and Alonso (22, 32) and Bui-Thanh et al. (25) demonstrated that the POD method could be used as a low-cost, low-order approximation to enhance the design process. The method proposed by LeGresley and Alonso is based on the gradient approach to cost function optimization. In both cases, conventional CFD methods were used to generate the data ensemble for the aerofoil inverse design problem. The POD procedure is then used to compute a set of optimal eigenfunctions from these snapshots. The two methods differ from each other in the way the cost function is evaluated and the way the optimal solutions are arrived at. In particular, LeGresley and Alonso have used the POD technique for both direct and inverse aerofoil design problems. In that work, a set of pressure field distributions corresponding to different aerofoil profiles were computed using an Euler solver. Different aerofoil profiles were created by perturbing the design variables of the base shape. The POD basis is then computed and used to construct a reduced-order model for Euler equations to compute new, approximate solutions for any arbitrary aerofoil at significantly lower computational costs. In this manner, both direct and inverse aerofoil design problems can be done efficiently using a gradient-based optimization procedure with the information from the reduced-order model. Bui-Thanh et al. applied the POD method to the problem of inverse aerofoil design. In that work, a collection of snapshots was generated by choosing a set of aerofoil shapes and computing their corresponding surface pressure distributions. The POD technique was then used to determine the optimal aerofoil shape that produces a given target pressure distribution. An important point of this method is that rather than containing only the flow variables, the snapshots are augmented to contain also the aerofoil co-ordinates. The minimal solution of the cost function is sought with a target vector that contains the required target pressure distribution and the unknown corresponding aerofoil co-ordinates. Thus, the target vector contains both known and unknown elements and so the procedure of Everson and Sirovich is used to reconstruct the missing data points.



## 2.2 Proper Orthogonal Decomposition

---

However, one of the most significant challenges is the use of POD-based reduced order models in high-speed flows with parametric variation. Indeed, as the shape/boundary condition parameter changes, the shock waves moves and classical POD/ROM techniques, which work well for subsonic flows, no longer provide reliable and accurate predictions. Lucia (33, 34) proposed and used a technique to exploit POD for accurately treating moving shock waves. This technique involves the decomposition of the solution domain to isolate those regions that contain shocks and so produces internal boundaries within the flow-field between the various domains. The main idea behind this is to use the POD modes for a global approximation of the flow field and to use a standard finite volume scheme in the region where the shocks occur. A reduced-order model for each region is developed independently and the solution for the entire domain is formed through a linking of the boundaries of each region, using optimization based solvers to ensure a smooth solution between overlapping parts of the internal boundaries. This technique was applied to a one-dimensional quasi-steady nozzle flow-field by Lucia for demonstration. However, LeGresley and Alonso (32) applied this technique for the shape optimization of a two-dimensional aerofoil. The results attained are good though some discrepancy could still be detected between the high-fidelity solution and the POD/ROM with domain decomposition.

Buffoni et al. (35) discuss three possible methods to adapt domain decomposition to transonic flows with shocks. The first is based on a Schur iteration where the solution of the low-order model is obtained by a projection step in the space spanned by the POD modes. The second is in the same spirit but instead of a DirichletNeumann iteration they employ a DirichletDirichlet iteration in the frame of a classical Schwartz method. The last approach is of different nature since the solution of the low-order model is not simply based on a projection in the space of the POD modes. It takes into account in a weak sense the governing equations by minimizing the residual norm of the canonical approximation in the space spanned by the POD modes. The main application is about the compressible Euler equations in a nozzle. The authors observe that the obtained results depend to a large extent on the database used for the POD modes. The locality of the approach is recognized, meaning that the approximation error can be large when the reconstructed solution is far from the training ones in the parameter space. They also point out that a major limitation of their method lies in the non availability of an efficient method to improve the approximation quality. Indeed, they conclude that, in

## 2. LITERATURE REVIEW

---

order to get better results, it is fundamental to increase the approximation accuracy by enriching the functional space in which the solution is sought, based on some objective in-fill criteria.

Toal et. al (36) proposed a POD-based re-parameterization for optimization purposes. This strategy, termed geometric filtration, was found to outperform a traditional kriging-based optimization, producing better designs for a considerable reduction in overall optimization cost. The geometric filtration strategy applies an initial kriging response surface model optimization to the original problem. From the results of this optimization a number of good design points are selected to form a snapshot ensemble for the purposes of POD. The POD basis functions then act as a re-parameterization of the original problem, filtering out badly performing designs and reducing the number of variables. A secondary kriging response surface based optimization is then carried out in which the modal coefficients of the POD bases are optimized. The optimization of a transonic airfoil for minimum drag to lift ratio was used as a test case to compare the geometric filtration strategy to a traditional kriging based optimization and an extensive direct optimization using a genetic algorithm. The traditional kriging strategy achieved 76.3% of the improvement obtained by the genetic algorithm but with only 300 objective function evaluations. However, applying geometric filtration to the same problem, again using 300 objective function evaluations, produced designs achieving 84.1% of the improvement obtained with the genetic algorithm, a substantial improvement over the traditional kriging strategy.

In a more recent paper, Braconnier et al. (37) combined steady compressible RANS equations, a POD reduced-basis method and a leave-one-out adaptive sampling technique. The proposed strategy was tested on an analytic test case and on the two-dimensional turbulent flow around a RAE2822 airfoil. It was shown that the adaptive resampling led to a higher speed of convergence with respect to classical Latin Hypercube *a-priori* design of experiments. However, the method applicability is not demonstrated on a real shape optimization problem, but just on a two-parameter Mach-angle of attack design space.

POD has been also investigated in Multi-disciplinary analysis and design. Lieu et al. (38, 39) applied POD-based ROMs for an aero-elastic analysis of a complete F-16 aircraft configuration with clean wings for varying Mach number and low angles of attack. In that work it has been shown that the POD method produces accurate ROMs

for the aero-elastic analysis of a complete aircraft configuration at a fixed flight condition. However, changes in the Mach number or the angle of attack often require the reconstruction of the ROM in order to maintain accuracy. Consequently, this destroys computational efficiency. In that work it was shown that *straightforward approaches for ROM adaptation lead to inaccurate POD bases in the transonic flight regime*. Thus, a new ROM adaptation scheme is proposed and evaluated for varying Mach number and angle of attack. This scheme interpolates the subspace angles between two POD subspaces and then generates a new POD basis through an orthogonal transformation based on the interpolated subspace angles. This computational methodology is applied to a complete F-16 configuration in various air-streams. The predicted aero-elastic frequencies and damping ratio coefficients were compared with counterparts obtained from full-order non-linear aero-elastic simulations and flight test data. Good correlations are reported in the transonic flow regime. It is reported that this technique has a significant potential for accurate, real-time, aero-elastic predictions.

## 2.3 Progress beyond the State of the Art

With respect to the overview of state-of-the-art methods and techniques in physics and non physics-based surrogate modelling, the present research aims at providing some advances and opening further questions on surrogate-assisted aerodynamic shape optimization of transonic aircraft.

### 2.3.1 Progress in geometry parameterization

The very first step when approaching a shape optimization problem is building a correspondence between the design variables, which are real or integer numbers controlled by the optimizer or decision maker, and the shape, which is typically a surface or a set of surfaces. However, common parameterizations, when exploring some regions of very large design spaces, often incur in unfeasible design candidates, where the source of unfeasibility is not related to constraint violation, but rather to unnatural, not realizable or exotic geometry. From another point of view, the designer is more and more interested to virtually reproduce and analyse any sort of shape, at least in a first, deeply explorative phase. Hence, the ideal parameterization is required to generate smooth and continuously varying shapes, hopefully controlled by parameters which are

## 2. LITERATURE REVIEW

---

well known to the designer. The contribution of the present research to this topic is given in chapter 3, where the CST parameterization is introduced and coupled to a CAD system able to generate the aircraft surfaces in a suitable format: the combined approach enhances the geometry parameterization through easy-to-use, analytic, fast, reliable, suitable to reproducing aircraft shapes and accurate methodology.

### 2.3.2 Progress in POD-based surrogate modelling

Another original contribution is related to the application of a Proper Orthogonal Decomposition technique as a surrogate model of aerodynamic flow fields under shape modifications and shock wave movement. POD has been generally used on fixed grids, so that only the aerodynamic variables are used to build the approximation. This approach is valid until the shape of the body is not modified and external flow parameters, like Mach number, angle of attack and so on, are sampled to generate POD snapshots. But, in a shape optimization problem, the geometry, and hence the mesh laying upon it, is changing with the design parameters and the relation between aerodynamic flow change and grid points movement represents a big issue. The present proposal is to solve it by including the grid coordinates in the snapshots definition and treating them as field variables as well as pressure, density, velocity and so on. Indeed, by employing a mesh generator with fixed block topology, the grid coordinates have almost the same properties of aerodynamic variables, as their amount of change decrease as the distance from the body increases.

### 2.3.3 Progress in transonic flow shape optimization

Moreover, dealing with transonic flow conditions, the shape optimization problem offers another fundamental question: how to deal with the movement of the shock wave and the variation of its intensity with the design parameters? As no investigations have been proposed in the literature on the specific topic (POD-based shape optimization in transonic flow), this represents a genuine original contribution, as a mixed zonal POD/CFD approach is introduced to partially get rid of discontinuities. The idea is to get a reduced order model of the aerodynamic flow where the shock wave is damped, i.e. in an outer domain sufficiently far from the body, and use this prediction to derive a physical boundary condition for the CFD flow solver to accurately compute the aerodynamic flow in the inner domain. Besides, the present insight into POD-based

surrogate modelling for transonic flow optimization allows to also provide an innovative contribution in the training phase. The quality of the prediction, and hence of the optimal search, highly depends on the dataset used to build the POD approximation, as the POD is as good as the feeding data. Indeed, the issue of how many computations should be performed to train a high-performance surrogate model has been already studied in the past, but the equally fundamental question about how to choose the training samples has marginally been investigated. Thanks to the joint work with Belgium research center CENAERO within an European Community funded project, some innovative techniques will be proposed in chapters 3 and 4 aimed at refining the surrogate model through adaptive Design of Experiment sampling based on POD model quality measures both in an off-line process, i.e. before the real optimization process, and in an on-line mode.

## 2. LITERATURE REVIEW

---

# 3

## Ingredients for aerodynamic design optimization

An aerodynamic design optimization task usually consists of an iterative process. Even if various approaches have been proposed in the past with different levels of complexity and innovation, the basic ingredients can be summarized as follows:

- choice of the design parameter set;
- parameterization of the geometry, i.e. translation of design parameters into shapes;
- surface/volume mesh generation, i.e. discretization of the shape and of the surrounding volume;
- CFD analysis, i.e. computation of the flow field variables in the discretized volume and onto the surface;
- computation of the objective function, usually given as a combination of integral quantities (like lift, drag, pitching moment) and extracted from the CFD analysis;
- return of the objective function to a minimization/search method (stochastic or deterministic approaches, with or without the computation of the derivative of any order) and identification of a new set, or a group, of design parameters;
- go back to the parameterization.

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

In this chapter, a general overview of our design optimization process is presented, as it configures as the core architecture which the surrogate/reduced order optimization is built upon in the following chapters. A particular focus is given to the parameterization approach through the coupling of the Class Shape Transformation (CST) method with a CAD system and to the governing equations for solving the aerodynamic flow. A state of the art optimization study is finally presented as an application example of the overall design process.

#### 3.1 An evolutionary optimization tool for aerodynamic design

Several techniques are today available for design through numerical optimization; concerning in particular the field of aerodynamic design, beyond methods developed *ad hoc* and characterized by inverse design capabilities, the techniques more properly related to direct optimization include mature gradient based methods, including more recent approaches like automatic differentiation and control theory based methods, and genetic algorithms. Generally speaking, it is not possible to state the superiority of one method over the others, if not with reference to a specific problem that needs to be faced. The important features that need to be evaluated are numerous:

- generality of the formulation vs. dependence on the problem;
- robustness vs. the need of human interaction;
- capability of multiple objective optimization vs. single-objective one;
- computational efficiency vs. the need of large computational resources.

From this point of view, the choice of one particular optimization technique implies to give up some possible advantages in favour of some others. On the other hand, due to the fact that aerodynamic shape design represents only a part of the overall design of a flying vehicle, and that the need for an effective multidisciplinary approach to the design task is arising, it is important for an optimization tool to combine as much as possible all the favourable characteristics stated above while avoiding the shortcomings.



#### 3.1.1 Optimization and design concepts

An aeronautical optimization system is an automatic procedure that starts from a given configuration and looks for an improved one. In order to do that, *design variables*, which can measure the absolute position in the design space or the changes with respect to the given configuration, and an *objective* function, which allows to measure the improvement of the configuration, must be defined.

From a theoretical point of view, the design variables represent the coordinates of a point in a sub-set of the real space in  $N$  dimensions  $\mathbb{R}^N$ , where  $N$  is the number of design variables. Integer or discrete design variables can also be employed, for example to map different topological settings of the configuration under analysis. We assume that an aeronautic configuration can be associated to each point of such a set, and an objective function can be evaluated for each configuration. With this assumption, a correspondence exists between the space of the design variables and the objective function, and points, which correspond to the absolute minimum of the objective, can be located. We call *optimum* the configuration(s) which is (are) associated to the minimum of the objective function.

*Constraint* functions can also be defined and evaluated, in the same way as the objective; by convention those points of the design space, which correspond to positive values of at least one constraint function, are taken out from the set.

Our optimization system is based on three main procedures: the optimizer, the configuration generator, and the objective and constraint functions evaluator.

From the mathematical point of view, the optimizer is an algorithm that looks for the absolute minimum of a function, which is defined in a sub-set of  $\mathbb{R}^N$ . When this function is continuous, together with its first derivatives, gradient-based optimization algorithms can be adopted. In case that no information is available concerning the derivatives of the objective, or in case that the definition domain is multi-connected, then a stochastic, gradient-free approach (e.g., genetic algorithm) is more appropriated.

Constraint functions can restrict the searching region by setting boundaries inside the domain, and in this way they can help the optimizer (particularly when a gradient-based method is adopted, which is capable to locate and follow the boundaries); on the other hand, boundaries could also dissect the definition domain, creating a multi-connected region, which cannot be handled anymore by a gradient-based optimization

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

method.

The configuration generator is a procedure that associates a point of the design variables space to an aeronautic configuration and produces all the information required to execute the evaluation procedure.

The technique adopted here is based on the concept of a *reference* configuration, i.e. the configuration generator starts from a given (*reference*) configuration, which corresponds to a specific point of the design variables space (the usual convention is to consider as *reference* the configuration corresponding to the design variables all equal to 0), and modifies it according to the modifications of the design variables. When the values of the design variables are modified, the modifications are reflected to some of the physical parameters which define the aeronautic configuration. Links between the physical parameters and design variables are set in the phase preliminary to the optimization, named *Problem Definition* (see par. 3.1.2). For each configuration a unique value of each objective and constraint functions must exist.

During the evaluation procedure the configuration is analyzed, in order to compute the objective function and the constraints. For aerodynamic analysis a grid generator and a CFD solver are required, together with tools for post-processing the solver output data. Geometric analysis could also be required (for example to compute wing tank volume or spar thickness).

Each operational condition that has to be considered to evaluate the objective and constraint functions is called *design point*. When more operational conditions must be taken into account to evaluate the objective and the constraints, then we are performing a *multi-point* optimization.

Multi-point optimization can be performed in two substantially different ways. The first one combines each design point in a single objective function using different weights related to the importance given to each design point. Another approach is *multi-objective* optimization (40, 41) where several objective functions can be considered by introducing a ranking criteria so that the configurations can be classified into dominated and non-dominated. The procedure here adopted can be used both for single and multi-objective optimization.

It is worth to note that in the multi-objective case the final result is not a configuration corresponding to the absolute minimum of an objective function, but a set of candidate solutions (Pareto set of non-dominated solutions). Each solution optimizes

### 3.1 An evolutionary optimization tool for aerodynamic design

---

the design problem at different level of compromise, in the sense that it is not possible to improve one objective function without deteriorating one of the others. In this case the choice of the final configuration is made after the optimization phase by introducing further decision-making criteria.

#### 3.1.2 Problem definition

Before the optimization process begins, a preliminary phase must take place, in which a reference configuration is selected and the optimization criteria are chosen. We call this preliminary phase *Problem Definition*. The operations performed in this phase are described in the present section. They are listed below:

- Links between design variables and physical parameters setting
- Design point(s) definition
- Optimization algorithm selection
- Objective and constraint functions selection

#### 3.1.3 Configuration definition and geometry parameterization

In any aerodynamic design optimization process, the choice of the mathematical representation of the geometry of an aircraft or its component, together with the fidelity of the flow solver and the type of optimization algorithm, has a significant impact on the computational resources, the exploration of the design space and the feasibility/smoothness of the generated geometries. The parameterization method also affects whether a meaningful “optimum” is contained in the design space and if an optimum design exists, whether or not it can be found. Desirable characteristics for any geometric representation technique include:

1. well behaved and produces smooth and realistic shapes;
2. mathematically efficient and numerically stable process that is fast, accurate, and consistent;
3. requires relatively few variables to represent a large enough design space to contain optimum aerodynamic shapes for a variety of design conditions and constraints;

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

4. allows specification of design parameters such as leading-edge radius, boat-tail angle, airfoil closure;
5. provides easy control for designing and editing the shape of a curve;
6. intuitive and geometric interpretation.

The idea behind the Class-Shape Transformation (CST) method, developed at Boeing and made popular by Brenda Kulfan (42), is that, even though aircraft shapes can vary a lot, the geometric representation is obtained through basic components of the configuration (e.g., wing, fuselage, nacelle, pylon, tailplane, canard, and so on) and, for each of them, two types of shapes can be identified:

- Class 1: airfoil-like shapes, basically defined by sweeping/lofting/twisting an aerodynamic shape (e.g., an airfoil) over one or more directions. Examples are wings, helicopter rotors, turbomachinery blades, tails, canards, winglets, nacelles obtained by revolution.
- Class 2: body cross-section shapes, whose cross-sections do not have an aerodynamic aspect, like fuselages, rotor hubs and shrouds, channel, ducts, lifting bodies.

Both types of shapes are troublesome as concerns the mathematical description. Indeed, some of their intrinsic properties, like the round nose of airfoil-like shapes and the large curvature variations over the body cross-section surfaces, generally imply a non-analytic function representation. Consequently, a large number of coordinates may be typically required to describe either class 1 or class 2 types of geometries. Numerous methods have been devised to numerically represent class 1 airfoil type geometries for use in aerodynamic design, optimization, and parametric studies. Commonly used geometry representation methods typically fail to meet the complete set of the previously defined desirable features. The CST method proposes the concept of representing arbitrary three-dimensional geometries as a distribution of fundamental shapes. The method has been successfully applied to describe and reproduce a large variety of aircraft component shapes, from airfoils/wings to axi-symmetric nacelles and bodies of revolution.

---

### 3.1 An evolutionary optimization tool for aerodynamic design

The present approach consists in coupling the Class-Shape Transformation method with the open-source CAD tool named GEOM, which is part of the SALOME package (43). The CST method is able to provide a wide range of aeronautic shape with reduced effort and good accuracy properties, but in a stand-alone mode it is not useful when a design process based on the definition of CAD surfaces is addressed. Indeed, even if the obtained surface definition is completely analytical, it is quite complex to directly include it into a CAD system, either because one does not usually have access to the sources and underlying mathematics of the CAD system and because this kind of developments would require IT skills which goes beyond the scope of the present research. Moreover, the flexibility and customization properties which come with scripting capabilities, available with SALOME, make the integration work easier and generalizable.

#### 3.1.3.1 The Class-Shape Transformation

In the most general case, the aircraft external shape is a surface  $\zeta$ , hence defined by two parameters  $(\psi, \eta)$ . The CST method allows to get any shapes as a “factorization” of a class function, representing the type of geometry, common to all the shapes belonging to that class, and a shape function

$$\zeta(\psi, \eta) = C(\psi, \eta)S(\psi, \eta) \quad (3.1)$$

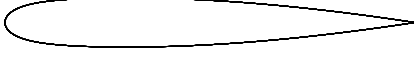
The shape function is a simple, well behaved analytic function with easily controlled key physical design features and inherent strong smoothing capability. The class function is introduced to generalize the methodology for applications to a wide variety of fundamental shapes. In the following sections, the detailed expression for class and shape functions, as in equation 3.1, and the corresponding shapes in terms of the cartesian components  $(x, y, z)$  are provided for an airfoil, a wing, an engine nacelle and a wing-nacelle pylon.

#### 3.1.3.2 Airfoil representation

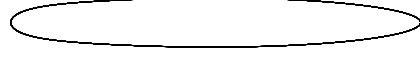
An airfoil can be represented as a curve, hence the surface parameter  $\psi$  is assumed as the x-coordinate along the chord line divided by the chord length  $L$  while the  $\eta$

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---



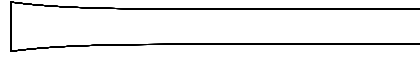
**Figure 3.1:** NACA-type airfoil,  $N1=0.5$ ,  $N2=1.0$



**Figure 3.2:** Elliptic airfoil,  $N1=0.5$ ,  $N2=0.5$



**Figure 3.3:** Biconvex airfoil,  $N1=1.0$ ,  $N2=1.0$



**Figure 3.4:** Duct shape,  $N1=0.001$ ,  $N2=0.001$

parameter is not defined. The general class function representing an airfoil-like shape is

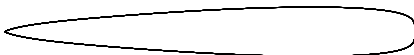
$$C_{N2}^{N1}(\psi) = \psi^{N1}(1 - \psi)^{N2} \quad (3.2)$$

In (42), it is shown that different combination of  $N1$  and  $N2$  mathematically defines a variety of basis shapes. Figures 3.1, 3.2, 3.3, 3.4, 3.5 and 3.6 shows some examples of general classes which can be obtain by tuning the two exponent of equation 3.2.

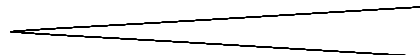
The class function defining round leading edge/pointed trailing edge airfoil shape is obtained with  $N1 = 0.5$  and  $N2 = 1.0$

$$C_{1.0}^{0.5}(\psi) = \sqrt{\psi}(1 - \psi) \quad (3.3)$$

The shape in figure 3.1 has been obtained by simply plotting the function in equation 3.3. By comparing it to the general expression given by the CST method in equation 3.1, we can observe that a unit shape function  $S(\psi) = 1$  already provides an airfoil shape. However, the unit shape function can be further decomposed into component airfoils: for example,  $S(\psi) = 1$  could be represented as the sum of  $S1(\psi) = \psi$  and  $S2(\psi) = 1 - \psi$ . By multiplying  $S1$  and  $S2$  by the class function, two airfoils are obtained and plotted in figure 3.7(a) and 3.7(b): one with a round nose and zero boat-tail angle, the other

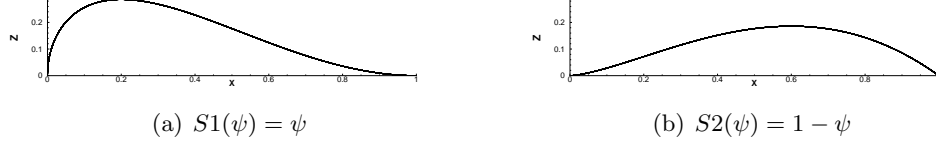


**Figure 3.5:** Low-drag projectile,  $N1=0.75$ ,  $N2=0.25$



**Figure 3.6:** Wedge airfoil,  $N1=1.0$ ,  $N2=0.001$

### 3.1 An evolutionary optimization tool for aerodynamic design



**Figure 3.7:** Decomposition of the shape function in two component airfoils

one with zero nose radius and a finite boat tail-angle. This is a direct result of the “partition of unity” property, which states that the sum of component terms is equal to one over the  $\psi$  interval  $[0,1]$ . Bernstein polynomials intrinsically satisfy this property in a scalable manner. Indeed, the Bernstein polynomial of any order can represent the unit shape function and, therefore, the individual terms in the polynomial can be properly scaled to represent a large variety of shapes. From a mathematical point of view, the unit shape function can be expressed as a Bernstein polynomial of order  $n$  ( $n + 1$  terms)

$$S(\psi) = \sum_{i=0}^n S_{i,n}(\psi) = \sum_{i=0}^n K_{i,n} \psi^i (1 - \psi)^{n-i} = 1 \quad (3.4)$$

where  $K_{i,n}$  are binomial coefficients

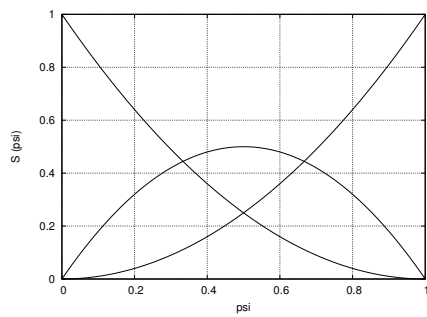
$$K_{i,n} = \binom{n}{i} = \frac{n!}{i!(n-i)!}$$

Figures 3.8(a), 3.8(b), 3.8(c) and 3.8(d) show some examples of Bernstein polynomial of various orders. The airfoil shape function is a smooth continuous and analytic function with finite derivatives of any order over the entire airfoil. Therefore, the Weierstrass theorem on continuous real-valued functions over compact intervals can be applied to derive interesting and desirable features of the proposed parameterization. In particular, for a given smooth shape, the CST method can represent it with any desired level of accuracy by increasing the order of Bernstein polynomials. Hence, the technique is able to catch the entire design space of smooth airfoils and every smooth airfoil is derivable from the unit shape airfoil or from any other smooth shape. The last property allows to not define a baseline airfoil in a shape optimization process as other common parameterization methods require.

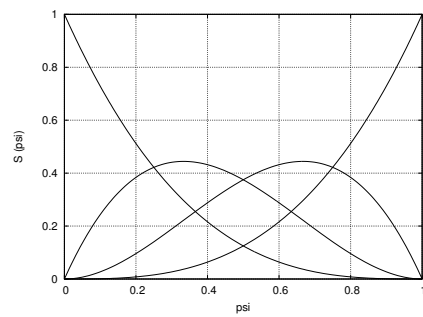
In (42) it is also shown that key features like leading edge radius, trailing edge thickness and boat tail angle are directly related to the bounding values of the shape

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

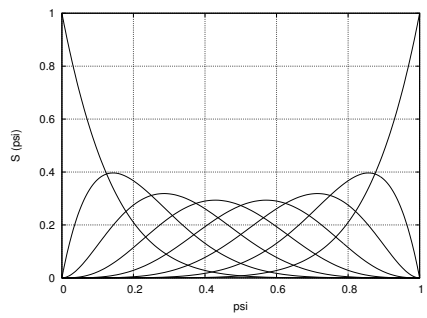
---



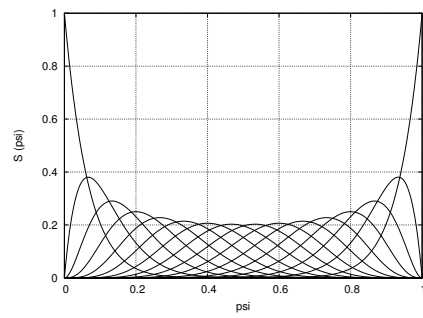
(a) 2nd order



(b) 3rd order



(c) 7th order



(d) 15th order

**Figure 3.8:** Bernstein component shapes



### 3.1 An evolutionary optimization tool for aerodynamic design

---

function. In particular, the value of the shape function at  $\psi = x/L = 0$  is a function of the leading edge radius  $R_{LE}$  in the form  $S(0) = \sqrt{2R_{LE}/L}$ . The value of the shape function at  $\psi = 1$  can be expressed in terms of boat tail angle  $\beta$  and trailing edge thickness  $\Delta z_{TE}$  as  $S(1) = \tan \beta + \Delta z_{TE}/L$ . Hence, when using the shape function transformation, controlling the endpoints of the shape function provides an easy and direct way to modify the airfoil nose and aft part. Once defined the general form of the shape function, it is easy to translate this information to the representation of the upper and lower surfaces of a cambered airfoil. Indeed, once selected the order  $n$  of the Bernstein polynomials, an arbitrary shape modification is obtained by properly scaling the component shapes through some coefficients that have to be determined depending on the objective of the study. In other words, the overall shape function equation for the upper and lower surfaces are

$$S_u(\psi) = \sum_{i=0}^n A_{u_i} K_{i,n} \psi^i (1-\psi)^{n-i} \quad (3.5)$$

$$S_l(\psi) = \sum_{i=0}^n A_{l_i} K_{i,n} \psi^i (1-\psi)^{n-i} \quad (3.6)$$

The coefficients  $A_{u_i}$  and  $A_{l_i}$  can be either defined *a-priori* in a design optimization process or a parametric shape variation or computed with a least-squares fit to match a specified geometry. Finally, an airfoil shape with rounded nose and finite trailing edge thickness can be expressed in cartesian component  $(x, z)$  as:

$$\frac{x}{L} = \psi \quad (3.7)$$

$$\frac{z_u}{L} = C_{1.0}^{0.5}(\psi) S_u(\psi) + \psi \frac{\Delta z_{TE}}{2L} \quad (3.8)$$

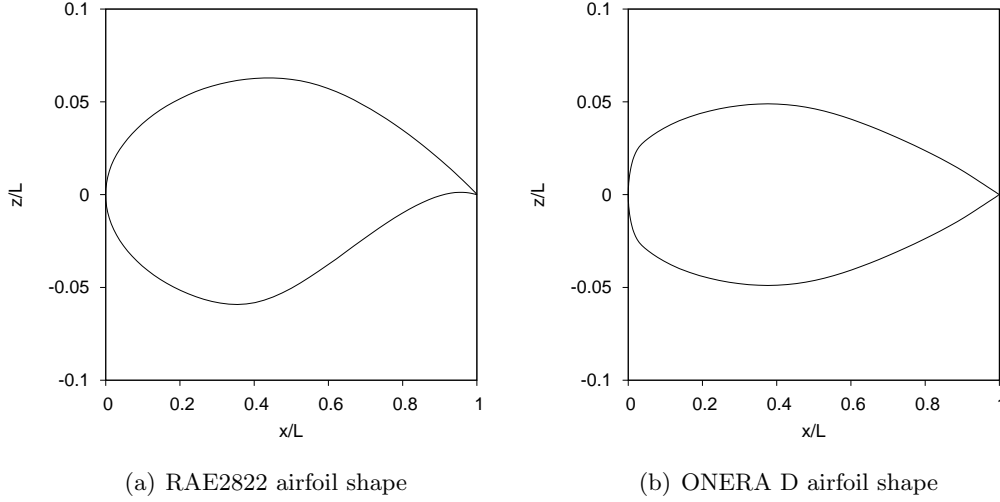
$$\frac{z_l}{L} = C_{1.0}^{0.5}(\psi) S_l(\psi) - \psi \frac{\Delta z_{TE}}{2L} \quad (3.9)$$

where  $c$  is the airfoil chord length and  $\Delta z_{TE}$  is the trailing edge thickness. If it is zero, the pointed aft end shape is obtained.

In order to test the CST approach and its convergence behaviour in representing specified shapes, two airfoils have been considered: the RAE2822 and the ONERA D airfoils. The shapes and the corresponding shape functions are depicted in figures 3.9(a)

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---



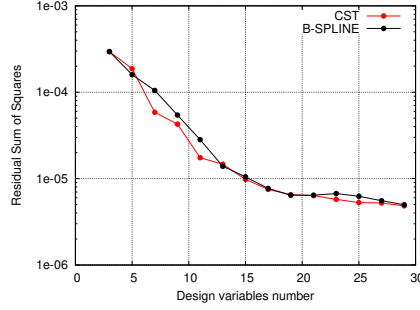
**Figure 3.9:** Test shapes

and 3.9(b). A scalable least-square problem has been set up in order to reconstruct the given shapes with different orders of the Bernstein polynomials. The RAE2822 and ONERA D airfoils are defined respectively by 129 and 143  $(x, z)$  coordinates. In order to perform fair comparisons between the “exact” and reconstructed shape, the  $\psi$  vector is based on the  $x/L$  coordinates of the given shape, so that only the  $z/L$  function will be compared. Given  $z$  the “exact”, specified coordinates vector of length  $m$  and  $z'$  the least-square minimizing one, assuming  $L = 1$ , the maximum residual norm  $RM$  and the residual sum of squares  $RSS$  will be used for comparisons:

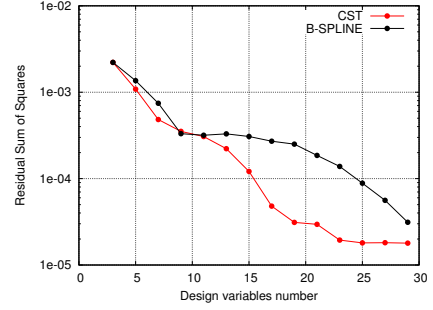
$$\begin{aligned}
 RM &= \max_j |z(j) - z'(j)| \\
 RSS &= \frac{1}{m} \sum_{j=1}^m [z(j) - z'(j)]^2
 \end{aligned} \tag{3.10}$$

Figures 3.10 and 3.11 report the obtained results as functions of the design variables number, which can be varied according to the order of the Bernstein polynomials. In order to assess the present parameterization efficiency to a well-established methodology, the same results obtained with a B-spline parameterization using the same number of

### 3.1 An evolutionary optimization tool for aerodynamic design

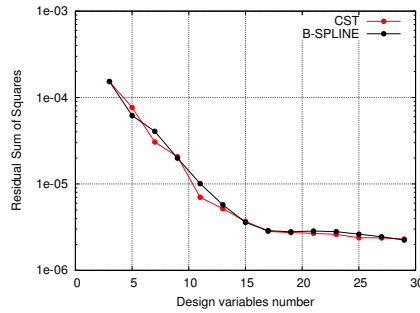


(a) RAE2822

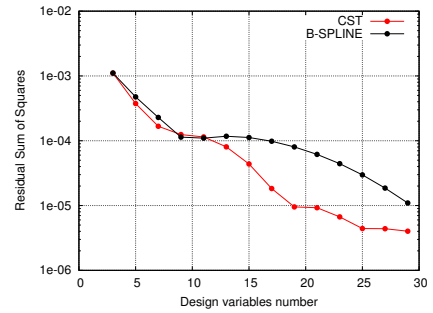


(b) ONERA D

**Figure 3.10:** Maximum residual error  $RM$  as a function of design variables number



(a) RAE2822



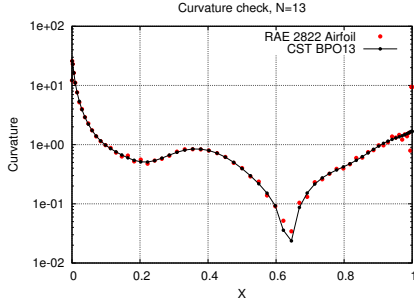
(b) ONERA D

**Figure 3.11:** Residual error sum of squares  $RSS$  as a function of design variables number

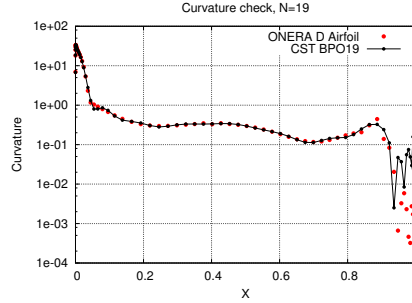
design parameters are shown. While on the RAE2822 airfoil, which has a more classical shape, the two methods show similar performances, with a strong drop of residuals with increasing design definition, the CST approach outperforms the B-spline approximation on the ONERA D airfoil.

Indeed, this airfoil was designed with a circular arc around the leading edge which is then joined to a classical airfoil in the remaining part. Hence, CST appears reliable and consistent when treating difficult shapes. This is also confirmed by the curvature analysis reported in figure 3.12. The curvature function reconstruction is compared only on the lower surface of the RAE2822 airfoil because the presence of the rear loading

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

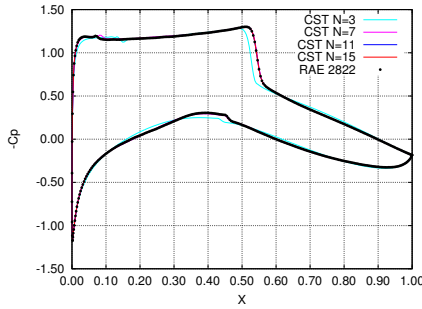


(a) RAE2822, lower surface, 14 design variables

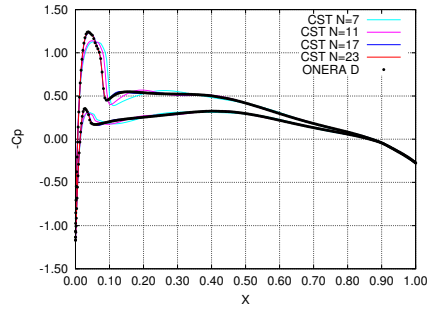


(b) ONERA D, 20 design variables

**Figure 3.12:** Airfoil curvature analysis



(a) RAE2822

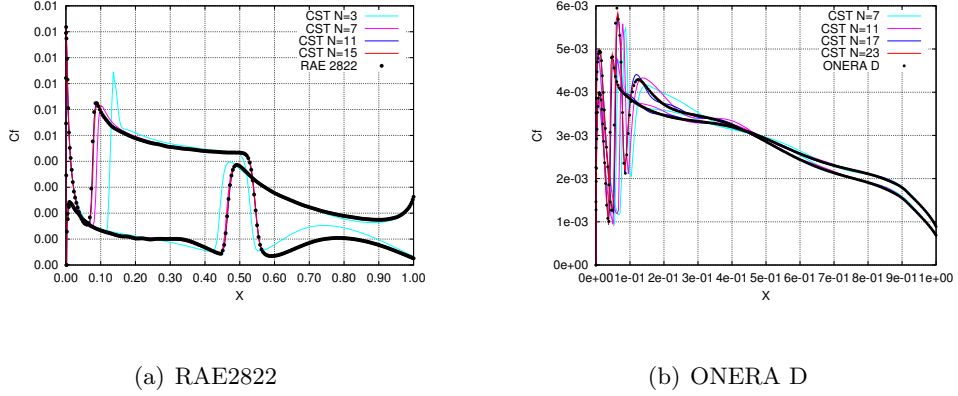


(b) ONERA D

**Figure 3.13:** Pressure coefficient analysis

makes it more difficult to catch than the upper one (a zero curvature point is present). The figure shows that 14 design variables are needed to accurately catch the RAE2822 curvature, while 20 are needed for the ONERA D airfoil. However, this last figure is affected by the original shape of the airfoil, which is not really smooth enough and presents some curvature oscillations in the rear part. In a design optimization process, the number of design variables could probably be much less, because we are interested in designing smooth shapes from the beginning and not in accurately reconstructing poor geometry data. Despite of this, the obtained approximation is completely satisfying also in the reconstructed cases.

### 3.1 An evolutionary optimization tool for aerodynamic design



**Figure 3.14:** Skin friction analysis

The last comparison can be done by using the most sensible tool for highlighting geometry defects: aerodynamics. An Euler-based viscid-inviscid interaction code is used to the purpose. The flow around the RAE2822 original and reconstructed shapes is computed at Mach number 0.73, Reynolds number  $6.5 \times 10^6$  and at an angle of attack of  $2.8^\circ$ . The flow around the ONERA D original and reconstructed shapes is computed at Mach number 0.75, Reynolds number  $10.5 \times 10^6$  and at an angle of attack of  $1.0^\circ$ . All the computations have been performed with free transition, which means that no boundary layer transition to turbulence is fixed *a-priori*, but it is computed by the solver through a stability analysis. Indeed, a defect in the geometry can trigger the transition at a completely different location, altering the whole prediction of the flow field around the shape. The obtained results in terms of pressure coefficient distribution and skin friction coefficient are shown in figures 3.13 and 3.14. It is clearly evident the refinement process which takes place by enriching the airfoil definition with the increase of Bernstein polynomials order. In particular, the flow characteristics on the reconstructed airfoils converge to the “exact” ones for both cases. A 7<sup>th</sup> order polynomial is enough to perfectly catch the RAE2822 aerodynamics, while a higher order (17<sup>th</sup>) is needed to obtain the same level of accuracy on the ONERA D airfoil.

In conclusion, the CST parameterization method shows interesting properties not only from a mere mathematical point of view, but especially in being a universal and powerful technique to define well-behaved aerodynamic shapes.

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

#### 3.1.3.3 Wing representation

A wing shape can be obtained by distributing the airfoil shape function along another surface parameter  $\eta$ , representing the wing span, with the desired law. Once selected  $N_x$ , the order of the Bernstein polynomial which represents the wing airfoils, the complete wing shape can be obtained by transforming the coefficients of the Bernstein polynomial into a distribution along the spanwise direction using any appropriate numerical technique. In this way, the entire wing surface is defined through the same basic component airfoils of the root airfoil, but the magnitude of each of them varies across the wing span according to the expansion technique. For example, using the Bernstein polynomial as expansion technique, the spanwise variation of each coefficients  $A_{u_i}$  in equation 3.5 can be stated in the form

$$A_{u_i} \equiv A_{u_i}(\eta) = \sum_{j=0}^{N_y} B_{u_{i,j}} S_{j,N_y}(\eta) \quad (3.11)$$

where

$$S_{j,N_y}(\eta) = K_{j,N_y} \eta^j (1 - \eta)^{N_y-j}$$

$N_y$  is the order of the expansion Bernstein polynomial and  $K_{j,N_y}$  is the usual binomial coefficient.

By inserting equation 3.11 into equation 3.5 and adopting the same technique for the lower surface, the bi-variate Bernstein polynomial shape function for the entire wing is derived

$$S_u(\psi, \eta) = \sum_{i=0}^{N_x} A_{u_i}(\eta) S_{i,N_x}(\psi) = \quad (3.12)$$

$$\sum_{i=0}^{N_x} \sum_{j=0}^{N_y} [B_{u_{i,j}} K_{j,N_y} \eta^j (1 - \eta)^{N_y-j}] K_{i,N_x} \psi^i (1 - \psi)^{N_x-i}$$

$$S_l(\psi, \eta) = \sum_{i=0}^{N_x} A_{l_i}(\eta) S_{i,N_x}(\psi) = \quad (3.13)$$

$$\sum_{i=0}^{N_x} \sum_{j=0}^{N_y} [B_{l_{i,j}} K_{j,N_y} \eta^j (1 - \eta)^{N_y-j}] K_{i,N_x} \psi^i (1 - \psi)^{N_x-i}$$

### 3.1 An evolutionary optimization tool for aerodynamic design

The graphical representation of basis functions  $S_{i,N_x}(\psi)S_{j,N_y}(\eta)$  is shown in figures 3.15(a), 3.15(b), 3.15(c) and 3.15(d) for  $N_x = 7$  and  $N_y = 3$ . The wing shape will be then represented by  $2 \times (N_x + 1) \times (N_y + 1)$  design parameters, namely  $B_{u_{i,j}}$  and  $B_{l_{i,j}}$ ,  $i = 0, \dots, N_x, j = 0, \dots, N_y$ . However, to assure continuity of curvature from the upper surface around the leading edge to the lower surface, the additional requirement  $B_{u_{0,j}} = B_{l_{0,j}}$  is introduced, so that the actual number of parameters becomes  $2 \times (N_x + 1) \times (N_y + 1) - N_y$ . By multiplying the wing shape function by the airfoil-like class function, the overall shape of the wing can be computed. However, a wing is generally and naturally conceived with spanwise distributions for twist angle, dihedral angle, sweep angle and taper ratio: this additional parameters have to be introduced in the CST model in order to cover realistic wing concepts. The actual wing surface cartesian coordinates can be obtained from the equations

$$x = \psi L(\eta) + x_{LE_{root}} + \int_0^\eta [\sin \Lambda(\eta)] d\eta \quad (3.14)$$

$$y = \frac{b}{2} \eta \quad (3.15)$$

$$z_{u,l} = L(\eta) C_{1.0}^{0.5}(\psi) S_{u,l}(\psi, \eta) + L(\eta) \psi [\Delta z_{TE}(\eta) - \tan \alpha_T(\eta)] + L(\eta) \int_0^\eta [\sin \delta(\eta)] d\eta \quad (3.16)$$

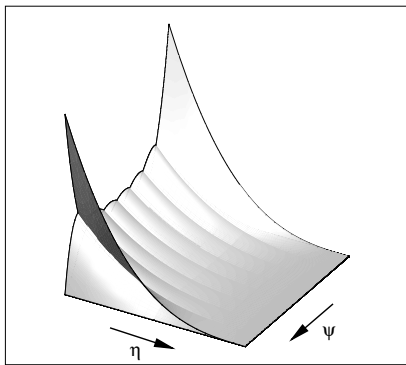
where the following spanwise distributions have been introduced:  $L(\eta)$  chord length,  $\Lambda(\eta)$  sweep angle,  $\Delta z_{TE}(\eta)$  trailing edge thickness,  $\alpha_T(\eta)$  twist angle,  $\delta(\eta)$  dihedral angle.  $b$  is the wing span length and  $x_{LE_{root}}$  is the x-position of the leading edge of the root airfoil. Several laws can be defined for spanwise distributions, but the standard approach which assures the manufacturing feasibility of the wing shape is to assume constant (e.g. sweep angle), piecewise constant (e.g. dihedral angle), linear (e.g. twist angle) or piecewise linear (e.g. chord length/tapering) variations.

#### 3.1.3.4 The integration within SALOME GEOM tool

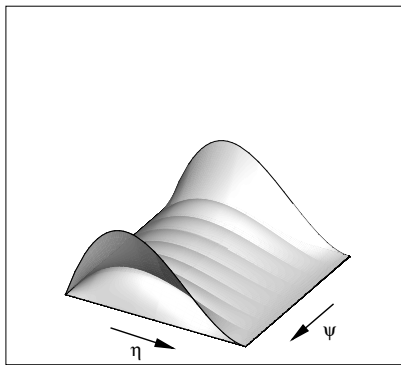
In order to provide a usable and interchangeable format for geometry definition, the CST wing parameterization has been coupled with a CAD system (GEOM) embedded within the open-source SALOME platform. Given a CST definition of a wing surface, the aim is to come out with a geometry file, typically in IGES or STEP format, which can be easily handled by common domain modeller/meshing tools, like ANSYS

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

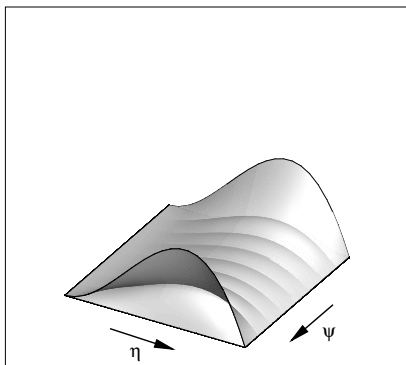
---



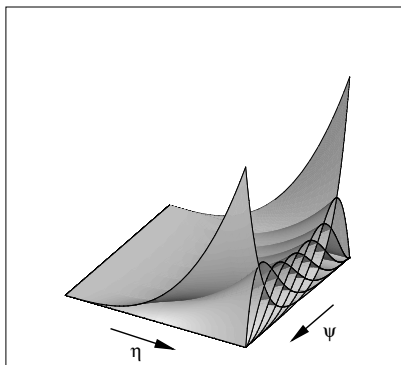
(a)  $S_{i,N_x}(\psi)S_{j,N_y}(\eta), i = 0, \dots, 7, j = 0$



(b)  $S_{i,N_x}(\psi)S_{j,N_y}(\eta), i = 0, \dots, 7, j = 1$



(c)  $S_{i,N_x}(\psi)S_{j,N_y}(\eta), i = 0, \dots, 7, j = 2$



(d)  $S_{i,N_x}(\psi)S_{j,N_y}(\eta), i = 0, \dots, 7, j = 3$

**Figure 3.15:** Bi-variate Bernstein polynomials



### 3.1 An evolutionary optimization tool for aerodynamic design

---

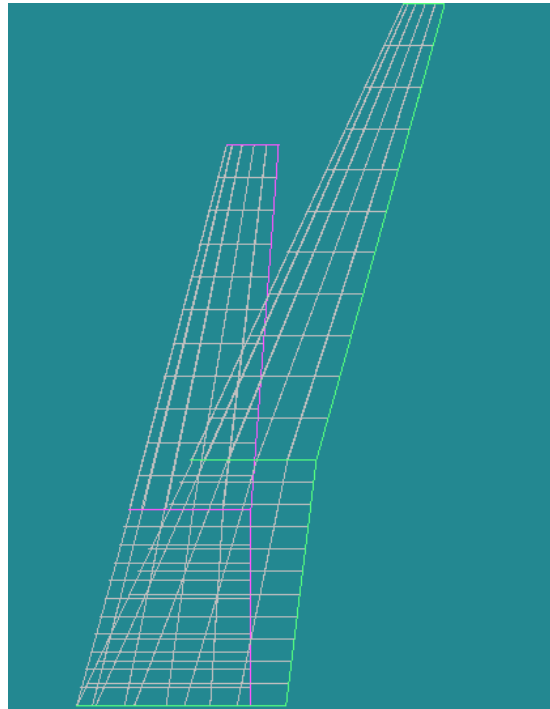
ICEM CFD, and usable within an automatic design process. The interface has been programmed by using the Unix Bash shell and Python scripting capabilities enclosed within the SALOME package. Given a set of design parameters, a scalable number of wing section shapes are derived from equations 3.14, 3.15 and 3.16 as a scalable set of coordinate points. This means that the wing shape can be virtually represented with an arbitrary number of sections, each described by an arbitrary set of points. The file of coordinates, together with information about the number of sections and the number of points per section, are read by SALOME GEOM and transformed into physical points. Then, each wing section is interpolated by means of a B-spline curve: in this process, the final curve has good properties of accuracy and smoothing thanks to the way the points have been defined, i.e. through the well-behaved, analytic CST geometric definition. The set of curves is then exploited to build two B-spline surfaces through a filling method, which means that each surface will pass through the defining curves. One surface is built on the inboard wing (from the root airfoil to the crank position), the other one on the outboard part (from the crank to the tip). Zero-th and first order continuity between two surfaces is assured by means of a sewing tool available in SALOME. Figure 3.16 shows a wing geometry generated with the CST method and shaped within SALOME GEOM tool together with a modified geometry obtained by using the automatic parametric variation of the wing shape: the new surfaces have been obtained by using the same process but setting a larger span length, a higher wing sweep angle and a lower taper ratio at kink and tip position. Figure 3.17 shows the smooth transition from root (thicker airfoils) to tip wing sections that can be obtained with CST representation: here, the bi-variate Bernstein polynomial is generated with  $8^{th}$  and  $2^{th}$  order to describe respectively the airfoil shape variation and its distribution along the span. In this case, the design variables have been assigned randomly, hence the given shapes are not so “aerodynamic”: however, the geometric modeller is able to define a smooth and continuous representation even for this case, proving to be a suitable approach for a fully explorative design process.

#### 3.1.4 Aerodynamic analysis

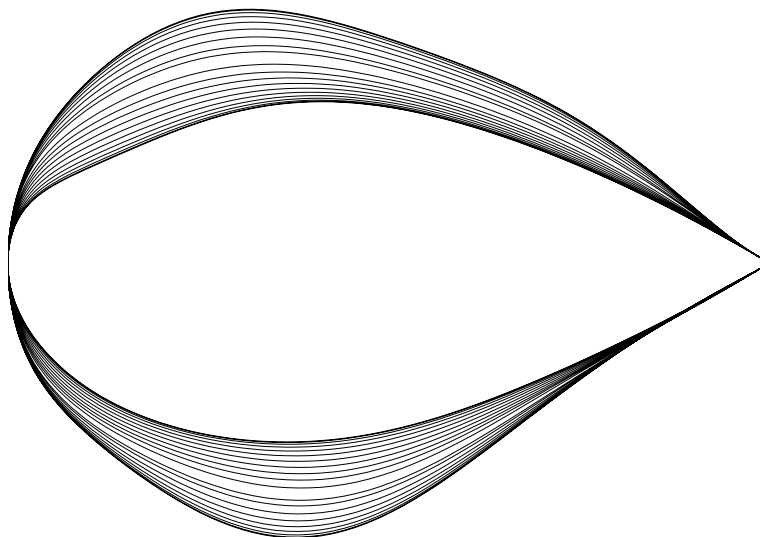
Once the master surfaces have been produced, they can be passed to a software tool for generating the computational grid. This task is conceptually the same for all kinds of grid generation. Different grid generators and solvers for CFD can be ideally selected.

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---



**Figure 3.16:** Wing shaping within SALOME



**Figure 3.17:** Smooth transition of randomly assigned wing sections along the span

### 3.1 An evolutionary optimization tool for aerodynamic design

---

In the case of multi-block, structured grid generation for aerodynamic evaluation, four sets of input data are needed (44, 45): the topology of the domain (i.e. how blocks are connected to each other), the geometry (i.e. the shape of surfaces which the boundary of the external blocks will lay on), the grid dimensions (how many cells have to be generated in each block) and the grid parameters (how cells have to be sized and clustered). Only the geometry of blocks, among those data, must change when design parameters are modified. A specific routine for automatic geometry generation has to be designed for each new type of configuration, when a different topology is required. It must be robust, so that hundreds of different geometries can be processed without failure, and the inspection of the configuration shape for debug purposes must be easy. The geometry generator reads in the master surfaces and writes out the geometry of multi-block domain in the proper format to be read by the grid generator. The geometry files produced by one routine are always associated with the specific topology file for which that routine has been designed. Routines to generate geometry files for airfoil, wing-alone and wing-body configurations are currently available.

Drag computation is very sensible to grid density, particularly when meshes are not very fine. Because of that, one of the issues in grid generation is the capability to produce grids with the same characteristics in terms of number of cells and density. The procedure helps to maintain a “standard” grid quality during the re-meshing of new configurations in the optimization loop.

Flow analysis is performed with a Navier-Stokes flow solver. Usually one solver run per design point is executed. Output files containing aerodynamic coefficients are processed for computing the objective and constraint functions.

#### 3.1.4.1 Governing equations

In this section, the fundamental mathematical model of Fluid Dynamics and the numerical solution model are introduced. The Navier-Stokes equations set is the most accurate mathematical model to describe the fluid flow motion. In the present context, they describe the motion of a perfect gas, i.e. the air, in absence of secondary phases and chemical reactions. The perfect gas obeys to two thermodynamic state equations: the first is the ideal gas law

$$p = \rho RT \tag{3.17}$$

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

which relates the gas pressure ( $p$ ), density ( $\rho$ ) and temperature ( $T$ ) through a gas constant  $R = 287 \frac{m^2}{s^2 K}$ . The second state equation expresses the dependency of the specific internal energy  $e$  on temperature only:

$$e = C_v T = \frac{R}{\gamma - 1} T \quad (3.18)$$

where  $C_v$  is the specific heat capacity at constant volume and  $\gamma$  is the ratio between the specific heat capacity at constant pressure and at constant volume.

The Navier-Stokes equations express the conservation/balance laws for mass, momentum and energy. Let  $\mathbf{u} = (u, v, w)$  be the flow velocity vector and  $E = e + \frac{|\mathbf{u}|^2}{2}$  the specific total energy. Under the hypothesis of *continuum* medium, no gas dissociation, no real gas effects, thermodynamic equilibrium, negligibility of body forces and heat sources, the Navier Stokes equations can be cast in a cartesian coordinate system as follows:

$$\frac{\partial U}{\partial t} + \frac{\partial F_1^c}{\partial x} + \frac{\partial F_2^c}{\partial y} + \frac{\partial F_3^c}{\partial z} = \frac{\partial F_1^v}{\partial x} + \frac{\partial F_2^v}{\partial y} + \frac{\partial F_3^v}{\partial z} \quad (3.19)$$

where

$$U = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix} \quad (3.20)$$

is the vector of the unknown flow variables,

$$F_1^c = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho uw \\ (\rho E + p)u \end{pmatrix} \quad F_2^c = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho vw \\ (\rho E + p)v \end{pmatrix} \quad F_3^c = \begin{pmatrix} \rho w \\ \rho wu \\ \rho wv \\ \rho w^2 + p \\ (\rho E + p)w \end{pmatrix} \quad (3.21)$$

### 3.1 An evolutionary optimization tool for aerodynamic design

---

are the convective fluxes and

$$\begin{aligned}
 F_1^\nu &= \begin{pmatrix} 0 \\ t_{11} \\ t_{12} \\ t_{13} \\ ut_{11} + vt_{12} + wt_{13} - q_1 \end{pmatrix} \\
 F_2^\nu &= \begin{pmatrix} 0 \\ t_{21} \\ t_{22} \\ t_{23} \\ ut_{21} + vt_{22} + wt_{23} - q_2 \end{pmatrix} \\
 F_3^\nu &= \begin{pmatrix} 0 \\ t_{31} \\ t_{32} \\ t_{33} \\ ut_{31} + vt_{32} + wt_{33} - q_3 \end{pmatrix}
 \end{aligned} \tag{3.22}$$

the diffusive fluxes. The stress tensor  $t_{ij}$  is related to the strain tensor through the molecular viscosity  $\mu$

$$t_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right) \tag{3.23}$$

with  $i = 1, \dots, 3$ ,  $j = 1, \dots, 3$  and the convention repeated indices summation is used. The heat flux  $q_j$  is defined by the Fourier law as

$$q_j = -\lambda \frac{\partial T}{\partial x_j} \tag{3.24}$$

By defining a set of reference quantities for length, density, temperature, velocity, viscosity and thermal conductivity scales ( $L_r, \rho_r, T_r, U_r, \mu_r, \lambda_r$ ), the dimension-less form of equations 3.19 can be derived, where the fundamental dimension-less Mach, Reynolds, Prandtl numbers define the relative importance of compressible, viscous and thermal effects

$$\begin{aligned}
 Ma &= \frac{U_r}{\sqrt{\gamma R T_r}} \\
 Re &= \frac{\rho_r U_r L_r}{\mu_r} \\
 Pr &= \frac{\mu_r c_p}{\lambda_r}
 \end{aligned} \tag{3.25}$$

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

In the laminar flow regime, the equations (3.19), once provided the initial and boundary conditions, form a closed system with the thermodynamic state equations and the additional relations  $\mu = \mu(p, T)$ ,  $\lambda = \lambda(p, T)$  between the molecular viscosity  $\mu$  and thermal conductivity  $\lambda$  and the thermodynamic properties of the gas.

In the turbulent regime, the scenario is different. The flow exhibits scales with large variations in space and time. The direct resolution of all the motion scales can be prohibitively expensive and depends on the Reynolds number. Following the Kolmogorov hypotheses, the statistics of the smallest scales of motion are uniquely determined by the molecular viscosity  $\nu$  and by the dissipation rate of the turbulent kinetic energy  $\varepsilon$ . The length, velocity and time Kolmogorov scales are built on the basis of dimensional analysis as

$$\eta = (\nu^3/\varepsilon)^{1/4} \quad u_\eta = (\nu\varepsilon)^{1/4} \quad \tau_\eta = (\nu\varepsilon)^{1/2} \quad (3.26)$$

with  $\varepsilon \approx u^3/L$ . The spatial resolution must be of order of magnitude  $\eta$  and the size of the computational domain has to be proportional to the most energetic scale of the flow  $L$ . The number of points required to resolve the Kolmogorov scales in the three computational directions is

$$N = N_1 * N_2 * N_3 = \left(\frac{L}{\eta}\right)^3 = \mathcal{O}(Re^{9/4}) \quad (3.27)$$

The equations have to be resolved in time with a time step  $\Delta t \approx \tau_\eta$  (without taking into account numerical stability requirements) for a number of time steps

$$N_T = \frac{T}{\Delta t} \approx \frac{L}{u\tau_\eta} = \mathcal{O}(Re^{1/2}) \quad (3.28)$$

The cost of a simulation is proportional to  $N * N_T = \mathcal{O}(Re^{11/4})$  rapidly growing with the Reynolds number.

The direct numerical simulation (DNS) of all the motion scales of a turbulent flow is limited to flows at  $Re = \mathcal{O}(10^{3,4})$ . For practical and industrial applications, an averaging of the Navier Stokes equations is performed in order to make affordable the numerical simulation of flows at higher Reynolds number.

#### 3.1.4.2 Reynolds Averaging of the Navier-Stokes Equations

A time averaging process of the (3.19) is performed. Instantaneous flow variables are considered as the sum of a mean and a fluctuating value :

$$f(\underline{x}, t) = \overline{f}(\underline{x}, t) + f'(\underline{x}, t) \quad (3.29)$$

### 3.1 An evolutionary optimization tool for aerodynamic design

The mean value is computed by averaging the variable over a time interval  $\Delta T$  much larger than the period of the fluctuating part but smaller than the time interval associated with the unsteady flow :

$$\bar{f}(\underline{x}, t) = \frac{1}{\Delta T} \int_0^{\Delta T} f(\underline{x}, t) dt \quad (3.30)$$

Therefore :

$$\overline{f'}(\underline{x}, t) = 0 \quad , \quad \overline{\bar{f}}(\underline{x}, t) = \bar{f}(\underline{x}, t) \quad (3.31)$$

but

$$\overline{f'(\underline{x}, t)g'(\underline{x}, t)} \neq 0 \quad (3.32)$$

The time averaging of (3.19), performed by applying the (3.29 - 3.30) taking into account the (3.31 - 3.32), leads to a system of equations for the mean value of the unknown flow variables (3.20). These equations, named Reynolds Averaged Navier Stokes (RANS), are formally identical to (3.21-3.22) with the exception of a new unknown term that comes from the convective fluxes. This term, the Reynolds stress tensor, is constituted by the double correlation of the turbulent velocity fluctuations :

$$\tau_{ij} = -\overline{\rho u'_i u'_j} \quad (3.33)$$

A set of transport equations to directly compute the components of (3.33) can be derived by multiplying the Navier Stokes equations by the velocity fluctuations and then time-averaging. The resulting Reynolds stress equations read, for an incompressible flow, as:

$$\begin{aligned} \frac{\partial \tau_{ij}}{\partial t} + u_k \frac{\partial \tau_{ij}}{\partial x_k} &= -\tau_{ik} \frac{\partial u_j}{\partial x_k} - \tau_{jk} \frac{\partial u_i}{\partial x_k} + 2\mu \overline{\frac{\partial u'_i}{\partial x_k} \frac{\partial u'_j}{\partial x_k}} + p' \left( \frac{\partial u'_i}{\partial x_j} + \frac{\partial u'_j}{\partial x_i} \right) + \\ &+ \frac{\partial}{\partial x_k} \left[ \nu \frac{\partial \tau_{ij}}{\partial x_k} + \overline{\rho u'_i u'_j u'_k} + \overline{p' u'_i \delta_{jk}} + \overline{p' u'_j \delta_{ik}} \right] \end{aligned} \quad (3.34)$$

New unknowns have been generated. Although equations for these terms could be obtained, the non linearity of the Navier Stokes equations would generate additional unknown terms. The usual approach is to relate the Reynolds tensor to the resolved mean flow variables through a turbulence model.

The Reynolds tensor, in analogy to (3.23), is made proportional to the mean flow strain tensor through the eddy viscosity :

$$\tau_{ij} = \mu_t \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_k}{\partial x_k} \delta_{ij} \right) - \frac{2}{3} \rho \kappa \delta_{ij} \quad (3.35)$$

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

where  $\kappa$  is the turbulent kinetic energy. The task on any turbulence model is to close the RANS equations by computing the eddy viscosity  $\mu_t$  that is assumed to depend on the velocity and length scale of the turbulent eddies

$$\mu_t \propto \kappa^{1/2} l^\alpha \quad (3.36)$$

Several turbulence models, ranging from algebraic to Reynolds stress models, have been developed and can be found in literature. In the algebraic models (46), the eddy viscosity is completely determined in terms of local flow variables. These models are cheap and robust, but are not able to take into account important effects of the flow history.

#### 3.1.4.3 The ZEN flow solver

The ZEN flow solver adopted for the RANS simulations is a multi-block well assessed tool for the analysis of complex configurations in the subsonic, transonic, and supersonic regimes (46, 47). The equations are discretized by means of a standard cell-centered finite volume scheme with blended self adaptive second and fourth order artificial dissipation. The pseudo time-marching advancement is performed by using the Runge-Kutta algorithm with convergence accelerators such as the multi-grid and residual smoothing techniques.

The turbulence equations are weakly coupled with the RANS equations and solved only on the finest grid level of a multi-grid cycle. Algebraic, one-equation, two-equations, and non linear eddy viscosity turbulence models are available.

**Numerical model** The Navier-Stokes equations (3.19), after applying the Gauss theorem, are written for each cell  $(i, j, k)$  of a computational domain as

$$\frac{d}{dt} \int_{V_{ijk}} U_{ijk} dV_{ijk} + \int_{\partial V_{ijk}} (F^c - F^v) dS_{ijk} = \int_{V_{ijk}} Q dV_{ijk} \quad (3.37)$$

where  $U$  is the vector of the unknown variables,  $F^c$  is the convective flux,  $F^v$  the viscous (physical and artificial) flux, and  $Q$  stands for the source term (if any). The volume of the computational cell is  $V_{ijk}$ .

The (3.37), by means of a cell centered finite volume approach, reduce to

$$V_{ijk} \frac{dU_{ijk}}{dt} + R_{ijk}^c - R_{ijk}^v - V_{ijk} Q_{ijk} = 0 \quad (3.38)$$



### 3.1 An evolutionary optimization tool for aerodynamic design

---

with  $R^c$  and  $R^v$  the total net fluxes ( convective and viscous respectively ) positive if outgoing from the volume  $V_{ijk}$ .

The residual  $R_{ijk}^c$  is obtained as the sum of the fluxes across the six faces of the cell  $(i, j, k)$

$$R_{ijk}^c = f_{i+1/2} - f_{i-1/2} + f_{j+1/2} - f_{j-1/2} + f_{k+1/2} - f_{k-1/2} \quad (3.39)$$

At the interface  $i + 1/2$  of the cell  $(i, j, k)$ , the flux  $f_{i+1/2}$ , positive if outgoing from the volume  $V_{ijk}$ , is evaluated as

$$f_{i+1/2} = \begin{cases} q_{i+1/2} \rho_{i+1/2} \\ q_{i+1/2} (\rho u)_{i+1/2} + p_{i+1/2} A_{i+1/2} \\ q_{i+1/2} H_{i+1/2} \end{cases} \quad (3.40)$$

where  $\rho_{i+1/2}$  is the density,  $p_{i+1/2}$  the thermodynamic pressure,  $(\rho u)_{i+1/2}$  the momentum, and  $H_{i+1/2}$  the enthalpy evaluated at the cell face by averaging between the values at the centers of the cells  $(i, j, k)$  and  $(i + 1, j, k)$ . The volume flux  $q_{i+1/2}$  is computed as :

$$q_{i+1/2} = \frac{(\rho u)_{i+1/2} \cdot A_{i+1/2}}{\rho_{i+1/2}} \quad (3.41)$$

where  $A_{i+1/2}$  is the area vector of the face  $(i + 1/2, j, k)$  pointing in the positive  $i$  direction.

The residual  $R_{ijk}^v$  is obtained as the sum of the fluxes across the six faces of the cell  $(i, j, k)$

$$R_{ijk}^v = g_{i+1/2} - g_{i-1/2} + g_{j+1/2} - g_{j-1/2} + g_{k+1/2} - g_{k-1/2} \quad (3.42)$$

The generic flux  $g_{i+1/2}$  requires, for the momentum equation, the evaluation of the velocities derivatives and of the heat flux for the energy equation.

The derivatives of the velocities are computed by integrating over a cell volume and applying the Gauss theorem. The gradient of the generic velocity component  $u$  is obtained as

$$(\nabla u)_{i,j,k} = \frac{1}{V_{i,j,k}} \sum_{f=1}^6 u_f \underline{A}_f \quad (3.43)$$

where  $u_f$  is the value of  $u$  at the face center, and  $\underline{A}_f$  is the area vector of the face.

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

Thus the derivative of  $u$  in the  $x_i$  direction results to be

$$\begin{aligned} \frac{\partial u}{\partial x_i} = & \frac{1}{V_{i,j,k}} \left( \frac{(u_{i+1,j,k} + u_{i,j,k})}{2} A_{i+1/2,j,k}^{x_i} - \frac{(u_{i,j,k} + u_{i-1,j,k})}{2} A_{i-1/2,j,k}^{x_i} \right. \\ & + \frac{(u_{i,j+1,k} + u_{i,j,k})}{2} A_{i,j+1/2,k}^{x_i} - \frac{(u_{i,j,k} + u_{i,j-1,k})}{2} A_{i,j-1/2,k}^{x_i} \\ & \left. + \frac{(u_{i,j,k+1} + u_{i,j,k})}{2} A_{i,j,k+1/2}^{x_i} - \frac{(u_{i,j,k} + u_{i,j,k-1})}{2} A_{i,j,k-1/2}^{x_i} \right) \end{aligned} \quad (3.44)$$

with  $A_{i+1/2,j,k}^{x_i}$  the  $x_i$ -component of the area vector of the face  $(i + 1/2, j, k)$

$$A_{i+1/2,j,k}^{x_i} = A_{i+1/2,j,k} n_{i+1/2,j,k}^{x_i} \quad (3.45)$$

where  $\underline{n}_{i+1/2,j,k}$  is the normal versor of the face.

The heat flux is computed as  $(\lambda_{tot})_{i+1/2}(\nabla_i T)_{i+1/2}$  where

$$(\lambda_{tot})_{i+1/2} = \frac{C_p \mu_{i+1/2}}{Pr} + \frac{C_p (\mu_t)_{i+1/2}}{Pr_t} \quad (3.46)$$

is the total heat conduction coefficient with  $\mu$  the molecular and  $\mu_t$  the turbulent viscosity and  $Pr$  and  $Pr_t$  the Prandtl and the turbulent Prandtl number respectively. The molecular and turbulent viscosity are computed by averaging between the cells sharing the considered interface

$$\mu_{i+1/2} = \frac{\mu_{i,j,k} + \mu_{i+1,j,k}}{2} \quad (3.47)$$

$$(\mu_t)_{i+1/2} = \frac{(\mu_t)_{i,j,k} + (\mu_t)_{i+1,j,k}}{2} \quad (3.48)$$

The  $i$  component of the gradient of the temperature  $T$  is evaluated as

$$(\nabla_i T)_{i+1/2} = \frac{T_{i+1} - T_i}{\Delta L_{i+1/2}} \quad (3.49)$$

where

$$\Delta L_{i+1/2} = \frac{V_{i,j,k} + V_{i+1,j,k}}{2|A_{i+1/2}|} \quad (3.50)$$

with  $|A_{i+1/2}|$  the area of the face  $(i + 1/2, j, k)$ , and  $V_{i+1,j,k}$  the volume of the cell  $(i + 1, j, k)$ .

### 3.1 An evolutionary optimization tool for aerodynamic design

---

The equation (3.38) is advanced in time by using a Runge Kutta (RK) algorithm. The  $m$  - stage formula, assuming that  $n$  is the known time level, is

$$U_{i,j,k}^{(0)} = U_{i,j,k}^{(n)} \quad (3.51)$$

$$(U_{i,j,k}^{(k)} - U_{i,j,k}^{(0)}) = \alpha_k \Delta t_{i,j,k} \left[ -\frac{1}{V_{i,j,k}} (R_{i,j,k}^c + R_{i,j,k}^v) + Q_{i,j,k} \right] \quad (3.52)$$

$$U_{i,j,k}^{(n+1)} = U_{i,j,k}^{(m-1)} \quad (3.53)$$

where  $\alpha_k$  is the RK coefficient and  $\Delta t_{i,j,k}$  is the time step which is evaluated for each grid cell separately. The convective residuals  $R_{i,j,k}^c$  are computed at each stage of the procedure, while the terms  $R_{i,j,k}^v$  and  $Q_{i,j,k}$  are calculated only at the first stage and then are frozen.

The use of a local time step does not influence the steady-state solution, and allows to have, where possible, larger time steps and thus to expel disturbances faster.

#### 3.1.5 Genetic algorithm - The ADGLIB library

In this section, the evolutionary computing method is presented as developed in the in-house optimization library ADGLIB. Genetic algorithms (GAs) belong to the class of evolutionary strategies, which common feature is the attempt to emulate the mechanisms typical of biological evolution. In biological evolution, genetic information stored in chromosomal strings evolves over generations to adapt favourably to a static or changing environment. In an analogous fashion, GAs are originally characterised by bit string representations of possible solutions to the problem at hand, and by mechanisms for the transformation and subsequent improvement of these coded solutions. Three basic components necessary for the implementation of a GA can be identified. The first one is a scheme that allows for a coded representation of possible solutions; as stated above, a bit string representation is usually adopted, by representing each design variable through a fixed length binary number, and linking together all the coded variables in a single string (chromosome); hence, each chromosome uniquely defines one solution (individual). Then, there must be a criterion for the evaluation of the fitness of each solution, allowing for a ranking of the individuals of the population; this criteria is of course problem dependent. The last component of the procedure is the most significant; it consists in those transformation functions that mimic the biological evolution process, when applied to a population of chromosomal representation of solutions to

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

the problem. A criterion for the selection of the pairs of individuals that are going to reproduce must be chosen, such that selection probability is higher for individuals characterised by higher fitness. Typical selection mechanisms are roulette-wheel and random-walk (48, 49); the differences between selection operators consist essentially in the level of selection pressure they realise, i.e. how much the selection probability is biased towards the best fit individuals. The two parents chosen are then mated through a crossover operator, which allows the recombination of their chromosomes; finally, the two strings obtained can undergo a mutation, consisting in a random variation of a little portion of the information coded in them. A number of different crossover and mutation operators can be used; the choice of these operators may have strong influence on the performances of the procedure, for a given optimisation problem (40). Thus, the fundamental steps that characterise a genetic algorithm can be summarised as follows:

1. a starting population of chromosome strings is first generated, usually in a random fashion or through a design of experiment technique or as a restart of a previous run;
2. each element of the population is evaluated by computing its fitness value;
3. the individuals to reproduce are selected, with bias allocated to the best fit ones;
4. new chromosome strings are generated by applying the crossover and mutation operators to the selected individuals, and a new generation is filled;
5. the evaluation, selection and reproduction phases are iterated and new populations are generated until a suitable solution is found.

As already stated, a peculiar feature offered by GAs is their capability to face multi-objective optimisation. When several design goals need to be achieved in an optimisation problem, these are usually combined together so that a single scalar objective function is obtained; this is generally achieved through a weighted linear combination of the different objectives, or on the basis of a demand-level vector. In this way, the problem becomes amenable to all classical optimisation algorithms. The drawback of this approach is that the solution of the problem is strongly dependent on the (arbitrary) choice of the relative weights assigned to the objectives; moreover, if the objectives to be minimised are of different nature, as happens for example when multi-disciplinary

### 3.1 An evolutionary optimization tool for aerodynamic design

---

optimisation problems are faced, it is difficult to understand how to interrelate them properly.

A different and well-established approach consists in classifying all potential solutions to the multi-objective optimisation problem into *dominated* and *non dominated* (Pareto optimal) solutions. The notion of domination can be defined as follows: let  $F = (f_1, \dots, f_n)$  the vector of a minimization problem with  $n$  objectives, and  $F^a, F^b$  be two candidates; we say that  $F^a$  dominates  $F^b$  if:

$$\forall i \in \{1, \dots, n\} \quad f_i^a \leq f_i^b$$

and

$$\exists i \quad | \quad f_i^a < f_i^b$$

The Pareto Front is the set of all the non dominated solutions; it follows from the definition that, if a solution belongs to the Pareto Front, it is not possible to improve one of the objectives without deteriorating some of the others.

By virtue of their structure, GAs are capable of facing multiobjective design problems in a more direct way; in fact, by selecting individuals according to the domination criteria instead of on the basis of a single fitness value, the set of Pareto optimal solutions can be discovered. In this way, a number of possible alternative solutions are obtained, each one meeting the requirements of the problem at different levels of compromise. Hence, the characterising feature of a multi-objective GA is the introduction of the Pareto criteria in the method used for individuals selection. In our system, this is accomplished through a random-walk operator: the current population is distributed over a toroidal landscape, a starting point is chosen at random, and the parents are selected as the locally non dominated individuals met in two subsequent walks, of a given number of steps, from that starting point; if more non dominated individuals are met, the first one encountered is selected.

The highlighted characteristics make GAs very attractive optimization tools, and explain the considerable growth of interest which has been devoted to them in recent years for applications of engineering interest. The major weakness of GAs lies in their relatively poor computational efficiency, as they generally require a very high number of evaluations of the objective function. For this reason, the use of GAs may become

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

un-practical when this evaluation is “expensive”, as happens for aerodynamic optimization applications where the solution of complex partial differential equation systems is necessary. This is the main reason why alternative techniques, like hybridization with gradients and use of surrogate and reduced order models, have been proposed to speed up the evolutionary optimization process.

#### 3.2 A state of the art optimization study

The challenging problem of wing design for small business aircraft configurations is here explored using evolutionary computing and the whole optimization process described in the present chapter. The study focuses on the aerodynamic analysis and optimization of a laminar wing in cruise and high lift conditions (50, 51). The main issue is the application of natural laminar flow technology which in turn implies pressure gradient optimization. As already mentioned above, evolutionary techniques, when applied to problems that require a substantial amount of computational effort for fitness evaluation, can be very expensive and prohibitive. Complementary techniques should be implemented to overcome these shortcomings. Nevertheless, even the most advanced and efficient approaches to design optimization through Evolutionary Computing still have some difficulties to be widely accepted and used in the day-by-day design practice of aerospace industry. Here, we want to show how an extremely challenging aerodynamic shape design problem can be effectively faced using quite simple single and multi-objective evolutionary optimization techniques in conjunction with a smart and well planned optimization strategy. Thus, a difficult three-dimensional aerodynamic shape design problem is split in two phases. In the first one a multi-objective design problem is solved on a simplified two-dimensional aerodynamic problem. This problem, as will result clearly in the following sections, is aimed to find a sub-optimal, but robust, configuration that can be safely used as a baseline for the subsequent three-dimensional single-objective optimization step. The multi-objective approach in this first step is essential, as the approximation errors deriving from the simplification of the computational model and the uncertainties in the real working conditions of the object to be designed, require a solution that does not decrease significantly the quality of its behaviour when working in conditions that are remarkably different from those for which it was designed. One of the techniques available to achieve this behaviour is

### 3.2 A state of the art optimization study

the resolution of a multi-objective problem in which the objectives are representative of an interval of working conditions. After this preliminary step, the designer will chose a particular solution from the Pareto front (possibly using some further discrimination criterion) to build the baseline configuration that will be used in the computationally intensive three-dimensional design phase. This last phase will therefore be much more devoted to refinement and exploitation rather than to the exploration of the design space. The work has been performed within the CESAR (Cost Efficient Small Aircraft Research) project (50), funded by the European Community within the 6<sup>th</sup> Framework Program.

#### 3.2.1 Problem Definition

Design points and constraints have been agreed and defined following the industrial guidelines and the data provided during the CESAR project. The aircraft operating points are summarized in Table 3.1. For the design purpose it was decided to focus on ‘Cruise Priority 1’ and ‘Low speed priority 4’, and, hence, the design task was conceived as a two-point problem where ‘Cruise 1’ is the main design point and the other one is treated as an additional one.

**Table 3.1:** Clean aircraft operating points

Priorities	Pressure altitude	True Airspeed	Reynolds number	Mach number
-	[ft]	[KTAS/ $ms^{-1}$ ]	Ref. to $C_r/C_t$	-
Cruise 1	35,000	400/205.8	$12.3 \cdot 10^6 / 4.6 \cdot 10^6$	0.694
Cruise 2	31,000	420/216.1	$14.6 \cdot 10^6 / 5.4 \cdot 10^6$	0.716
Cruise 3	41,000	350/180.0	$8.2 \cdot 10^6 / 3.1 \cdot 10^6$	0.610
Low sp. 4	0	90/46.3	$7.2 \cdot 10^6 / 2.7 \cdot 10^6$	0.136

A synoptic view of the aircraft design problem definition is given in Table 3.2. Constraints and objectives definition are cast in the presented form in order to give sense to the aerodynamic design problem. The maximum lift attainable at design point 2 is considered here as a constraint to satisfy. Hence, a “reference” value ( $=1.4$ , according to the specifications) is fundamental to impose because the optimizer has to search the best solution in the design space subject to a minimum production of lift at low

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

speed: indeed, an incontrovertible evidence and experience exists about the conflicting nature of transonic laminar flow enhancement and low speed performances. The key parameter is the leading edge airfoil shape: a small leading edge radius is desirable for laminar flow but at the same time it strongly deteriorates the stall characteristics.

#### 3.2.2 Optimization strategy

The design optimization of such a transonic wing under laminar flow conditions requires several design variables to control wing shape and twist angle distribution. In addition, the presence of the engine nacelle and the simulation of the inlet mass flow makes the geometry and the corresponding flow pattern complex to analyze. Another key issue is the estimation of the maximum lift coefficient at low speed by means of CFD methods which would enormously increase the computational cost for the aerodynamic evaluation. Thus, these features would imply a huge number of very long computer

**Table 3.2:** Optimization problem definition

Design variables	Wing section twist Wing section shape	3 sect., 3 des. var. 3 sect., 60 des. var.
Design point 1 <i>(Cruise Priority 1)</i>	Mach Reynolds Lref [m] Altitude [m]	0.694 9.0E+06 1.786 35000
Design point 2 <i>(Low speed priority 4)</i>	Mach Reynolds	0.136 5.27e+06
Constraints	Wing $C_L$ @ DP1 Wing $C_M$ @ DP1 Fuel Tank Volume Maximum $C_L$ @ DP2	[0.20 : 0.30] > -0.1 > 0.6205m <sup>3</sup> ≥ 1.40
Objectives	L/D @ DP1 Laminar extent <sup>a</sup> @ DP1	To be maximized To be maximized

<sup>a</sup> The laminar extent is computed as  $\tilde{X} = \frac{2}{b} \int_0^{b/2} X_{tr}(y) dy$ , where  $X_{tr}(y)$  is the locus of transition locations along the wing span and  $b$  is the wing span length. It represents the ratio between the laminar flow area and the total wing area.



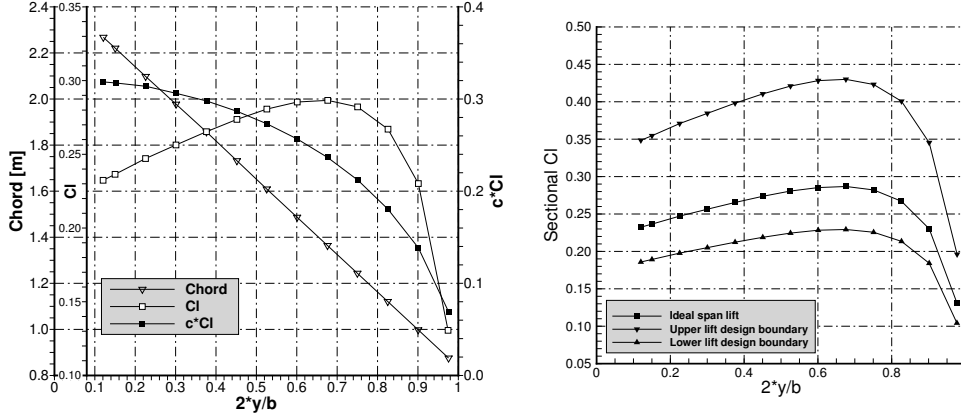
simulations in order to explore the design space. To reduce the overall computational cost, past researches and experience suggest to first design optimal wing section shape with reduced order methods. This approach is feasible because the problem under investigation deals with the low transonic Mach number aerodynamic flow around a low sweep angle wing; besides, industrial experience show that the wing aerodynamic behaviour in take-off and landing strongly depends on a properly designed airfoil shape. From this perspective, multi-point approach coupled to quasi-3D methods has proved to be one of the best compromise in laminar wing design: indeed, reliable 3D performances prediction, wide design space search and computational speed are matched at the same time. Hence, the optimization task has been split in two separate stages: a preliminary stage consisting in wing airfoil optimization and a final stage in which the wing shape is optimized by taking into account the fully three-dimensional flow around the wing, the fuselage body and the engine nacelle in working conditions.

#### 3.2.3 Wing section optimization with a 2.5D approach

The main objective of this preliminary design phase is to optimize selected wing section shape at and around ideal span load conditions. The idea is to get airfoil shapes which might maintain an optimal performance throughout a range of design points: this ranges are chosen to reproduce the ideal flow conditions onto the actual 3D wing. The aerodynamic method is based on a quasi three-dimensional or two-and-a-half-dimensional viscous/inviscid interaction approach able to solve coupled Euler + boundary layer equations (52, 53). The methodology is referred to as “2.5D” because it is basically two-dimensional, but it takes into account some aspect of three-dimensional wing flow such as sweep angle effect and boundary layer cross-flow.

In order to design efficient airfoils for a wing whose geometric and aerodynamic data are known, it is needed to compute the desired lift distribution as a function of the spanwise axis. Given the wing planform (and hence the wing area  $S$  and the chord distribution  $c(y)$  along the wing span) and the global lift  $C_L$  that the wing must generate, it is straightforward to get the corresponding ideal load distribution by simply searching for an elliptic function  $f(y) = c(y)C_l(y) = K_0\sqrt{b^2/4 - y^2}$  that satisfies the equation  $C_L = \frac{2}{S} \int_0^{b/2} f(y)dy = \frac{2}{S} \int_0^{b/2} K_0\sqrt{b^2/4 - y^2}dy$ . Solving for  $K_0$  gives  $K_0 = 8C_L/\pi AR$ , where  $AR$  is the wing aspect ratio. The distribution of sectional lift coefficient  $C_l(y)$  is then computed. Figure 3.18(a) shows the actual CESAR wing

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION



(a) Elliptic chord and lift coefficient distribution (b) Ranges of lift distribution for 2.5D airfoils

**Figure 3.18:** Sectional distributions along the wing span

planform chord distribution (black curve), the elliptic spanwise loading at  $C_L = 0.25$  (red curve) and the corresponding sectional lift distribution (blue curve). Three span sections ( $2y/b = 0.15, 0.50, 0.90$ ) are selected to define the wing shape, representing respectively the inboard, mid and outboard part of the wing. However, the parametric shape change is applied only to a single representative airfoil, which is then scaled to the thickness and flow conditions corresponding to the three selected sections. Hence, the global airfoil shape (i.e., thickness and camber distribution) will be the same throughout the wing. For each wing section, the curves plotted in figure 3.18(a) give the corresponding values for chord (and, hence, Reynolds number) and design lift coefficient. It is evident that one airfoil shape cannot be representative of wide span portions because chord and lift distribution vary quite rapidly along the wing. For this reason and, moreover, to allow extended optimal aerodynamic performances at incidence angles surrounding the cruise point (i.e., to cover slight off-design conditions and to take into account the effect of downwash on the effective sectional angle of attack), a multi-point approach is followed: two additional design points are defined for each wing section by simply scaling the lift coefficient by a factor 1.5 and 0.8. The obtained sectional lift distribution ranges are reported in Figure 3.18(b). A further design point is added to consider the airfoil low speed performances in clean conditions. The goal of this optimization stage is twofold:

### 3.2 A state of the art optimization study

- to minimize the wing section viscous drag by reducing the form drag and enhancing the laminar flow as much as possible;
- to keep the airfoil maximum lift coefficient higher than a minimum allowed level (the  $C_{l_{\max}}$  is treated as a constraint function).

Basically, the problem definition of the 2.5D design mirrors the contents of Table 3.2. Two set of GA-based optimization run have been performed, each sharing the same design variables and design points but with a different setting up of the optimization strategy.

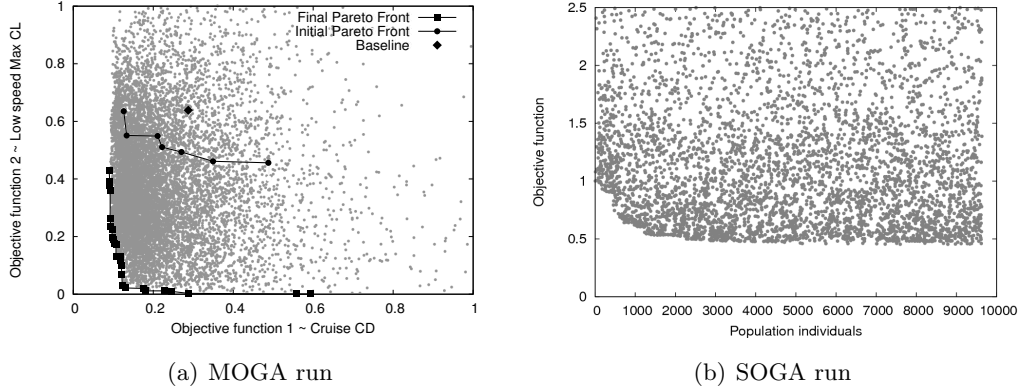
The first one, here referred to as MOGA (Multi-Objective Genetic Algorithm), is a bi-objective optimization: the contributions of cruise (Obj 1) and low speed (Obj 2) design points have been split in two separated cost functions each one in competition with the other and defined as follows:

$$Obj1 = \sum_{i=1}^3 \sum_{j=root}^{tip} \left[ aCd + b \left( \tilde{X}^{up} - \bar{X}^{up} \right)^2 + c \left( \tilde{X}^{lo} - \bar{X}^{lo} \right)^2 \right]_{ij}$$

$$Obj2 = \sum_{k=root}^{tip} \left[ d \left( Cl_{\max} - \bar{Cl}_{\max} \right)^2 \right]_k$$

Here, the double summation is introduced to count the contributions of the three sections which define the position along the wing span (inboard, mid, outboard airfoil) through index “j” and the three design points at and around the cruise conditions through index “i” (1 is for the cruise point, 2 and 3 are for the additional cases at higher and lower lift levels). The quadratic penalty functions are active only when the constraint is not satisfied, i.e. when the performance index  $(\tilde{X}^{up}, \tilde{X}^{lo}, Cl_{\max})$  is below the threshold (denoted by bar quantities); otherwise, their value is assumed as zero. The coefficients  $a, b, c, d$  are scaling factors used to adjust the order of magnitude, and, hence, the relative importance, of the objective function components. The MOGA optimization might be considered as an exploratory descent into the design space with the aim of uncovering trade-offs and compromise trends between cruise and low speed performances through the optimal Pareto front concept. A population of twenty-four individuals is evolved through five-hundred generations (a total number of twelve thousands cases were analyzed). The probability of crossover activation is eighty percent, while the rate of the mutation operator is 0.7 percent. The elitism strategy is

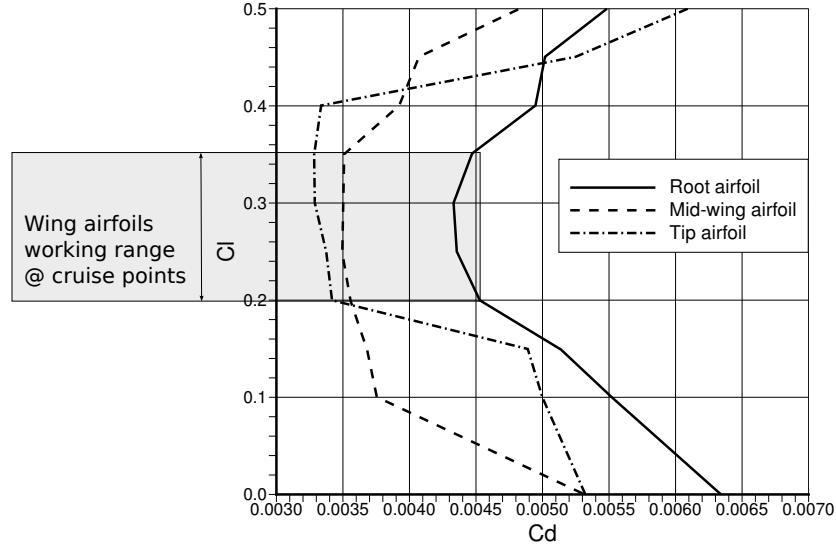
### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION



**Figure 3.19:** 2.5D optimization convergence histories

activated on the cruise related objective (Obj 1): this means that each newly generated population will include the individual which is the best fit with respect only to Obj 1. The main advantage of this approach is to assure that the bi-objective run would not produce a worse fit element, with respect to Obj 1, than a single objective run on Obj 1. Figure 3.19(a) reports the convergence history of MOGA run in the objective plane. Together with the fitness of the individuals along generations (red points), the plot also highlights the baseline fitness (yellow point), the Pareto front obtained at the end of the first generation (black line points) and the final Pareto front (blue line points). The push towards the “utopia” point (i.e., the ideal point where both the objective function values are zero) is clearly evident, although the obtained front cannot be considered converged to the true non dominated one.

The second step of 2.5D airfoil optimization is here referred to as SOGA (Single-Objective Genetic Algorithm), as only one objective function is minimized within the same problem definition by letting evolve a sub-optimal population. Indeed, the individuals composing the initial population are selected among the Pareto solutions of the previous MOGA run. Hence, the SOGA exploration can be considered as a further refinement stage. The objective function is here defined as a combination of high speed (cruise drag and transition location) and low speed performances (maximum lift) through weights and penalties. Its rather complicated expression now has the form:



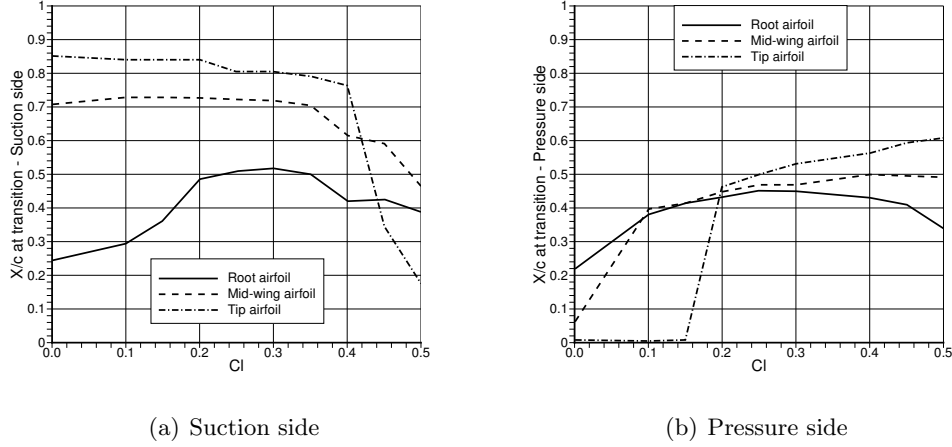
**Figure 3.20:** Wing section polar curves at cruise conditions

$$Obj = \sum_{k=root}^{tip} \left[ a' Cd_1 + b' \left( \tilde{X}^{up} - \bar{X}^{up} \right)^2 + c' \left( \tilde{X}^{lo} - \bar{X}^{lo} \right)^2 + d' \left( Cl_{max} - \bar{Cl}_{max} \right)^2 + e \left( Cd_2 - \bar{Cd}_2 \right)^2 + f \left( Cd_3 - \bar{Cd}_3 \right)^2 \right]_k$$

In particular, the drag coefficients for cruise points 2 and 3 ( $Cd_2, Cd_3$ ) are now treated as quadratic constraints whose threshold values ( $\bar{Cd}_2, \bar{Cd}_3$ ) are based on the best-fit elements of MOGA optimization. The coefficients  $a', b', c', d', e, f$  are scaling factors, as previously mentioned.

A population of thirty-two individuals evolved through three-hundred generations. The adopted genetic algorithm uses binary encoding with Gray code (54), and each design variable is coded using a 31-bit binary substring. The probability of crossover activation is eighty percent, while the rate of the bit-mutation operator is 0.5 percent. The GA parameters were chosen following the experience gathered in previous design problems similar in nature. Figure 3.19(b) shows the convergence history for the SOGA run: the objective function is reported on the y-abcissa and normalized with respect to the best fit individual of the initial population. Hence, a global improvement of 50% (in terms of the defined objective function) is obtained at the end of the SOGA optimization. Figure 3.20 shows the high speed polar curves for the optimized wing sections: the plot highlights the effectiveness of the optimization process as the minimal

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION



**Figure 3.21:** Transition prediction as a function of  $C_l$

values of drag coefficients are reached onto a well defined branch of the aerodynamic polar. Figure 3.21(a) and 3.21(b) report the transition location on the upper and lower airfoil side as a function of the lift coefficient: the stability of the boundary layer is reached and maintained over the specified design points. The optimized wing airfoil geometry is reported in Figure 3.22(a). The pressure coefficient curves are shown in Figure 3.22(b) (inboard section), Figure 3.22(c) (mid-wing section) and Figure 3.22(d) (outboard section). It can be noticed that, while the inboard and mid-wing airfoil maintain almost the same shape of the pressure distribution at different lift levels, the outboard airfoil presents some peaks on the lower side at lower lift cases. This is mainly due to the reduced airfoil thickness and to the particular shape of the leading edge. Figure 3.22(e) shows the comparison of the lift curve and maximum lift prediction between a RANS fully turbulent approach and the ESDU semi-empirical procedure (55) on the optimized mid-wing airfoil: the agreement is remarkable especially about the estimation of the maximum lift and of the lift curve slope decreasing. Figure 3.22(f) shows the low speed polars of the inboard, mid and outboard wing sections under the respective operating conditions: basically, the three curves differ for the wing airfoil maximum thickness and the Reynolds number, as the global airfoil shape is the same along the wing. A general decrease of the airfoil maximum lift is observable towards the wing tip due to reduced thickness and Reynolds number: a proper adjustment of

wing twist will help to recover this feature in the three-dimensional optimization phase.

### 3.2.4 Automatic 3D mesh generation

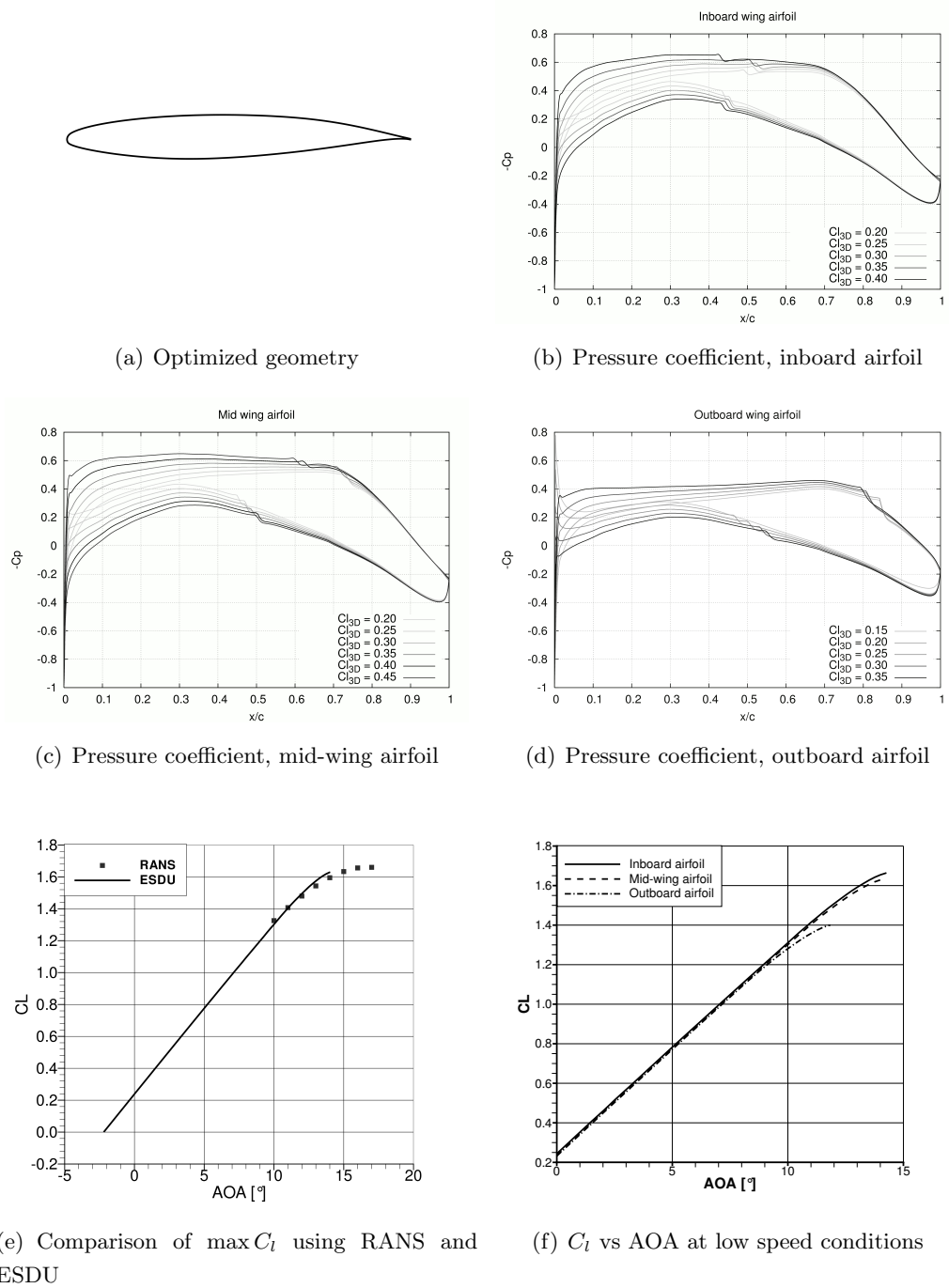
Before setting up the meshing strategy, a suitable and simpler CAD description of the geometry surfaces was obtained from the initial CAD draw provided by Piaggio (see figure 3.23(a)). This “cleaning” activity was needed due to the presence of many small surfaces (377 surfaces to describe the wing-body-fairing-nacelle-pylon configuration), bad connections and “leakage flows” between adjacent surfaces and trimmed surfaces. The CAD repairing was made through reconstruction, simplification and joining operations on each surface; connections between surfaces were restored through border curves sharing. The result of CAD repairing is shown in Figure 3.23(b) where the whole configuration was described with just 25 surfaces without any hole or overlap.

As the geometry is quite complex due to the presence of the nacelle body, particular attention has been given to the domain modelling and mesh generation phase in order to build a suitable structured block topology able to be easily applied to a wing shape optimization process. The software ICEM CFD has been used to generate structured volume grids around the wing-body-rear mounted nacelle configuration, and a batch script, having the form of a ICEM replay file, has been set up to automatically perform the re-meshing of a new configuration. The resulting grid size (about 10 million cells) is computationally very expensive to be used in an optimization run, so that a coarser grid level (about 1.2 million cells) is instead used which anyway is conceived to satisfy the requirement of unit order  $y^+$  (dimensionless wall distance) by controlling the height of the first cell off the wall. The fine mesh size will be used to validate the optimization results. Figure 3.24 shows the coarse surface mesh on the whole configuration: blue lines highlight interfaces between computational blocks.

### 3.2.5 Wing-body-nacelle optimization

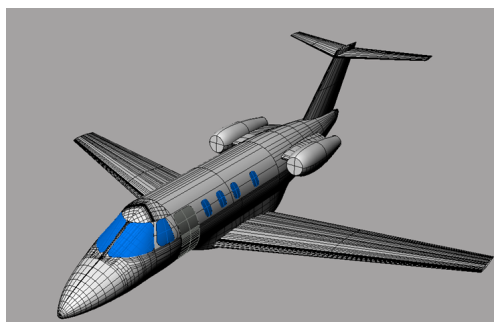
The final optimization stage has been conceived to include the engine mass flow in the simulation process and to take into account its effects on the wing laminar flow. Indeed, the fully three-dimensional flow around the wing, fuselage and rear-mounted nacelle is solved: the ZEN RANS flow solver is used in fully turbulent mode to estimate

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

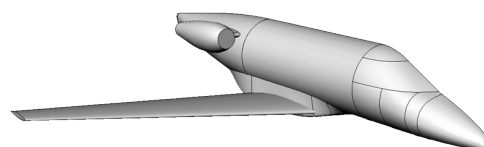


**Figure 3.22:** Airfoil optimization results



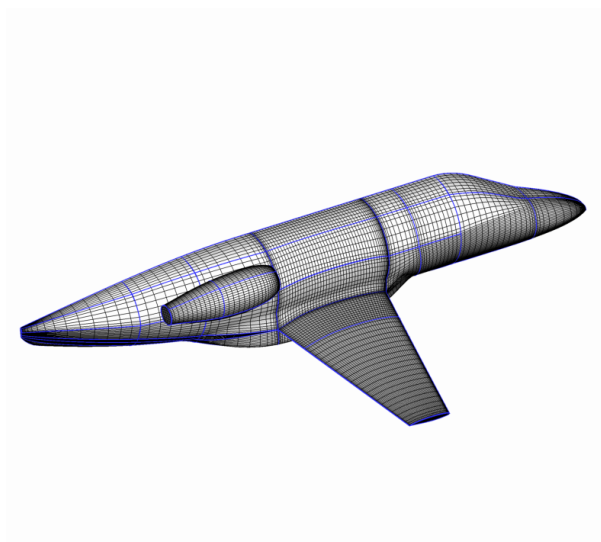


(a) Initial CAD definition



(b) “Cleaned” CAD drawing

**Figure 3.23:** The aircraft configuration



**Figure 3.24:** RANS skin mesh with block structure highlight

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

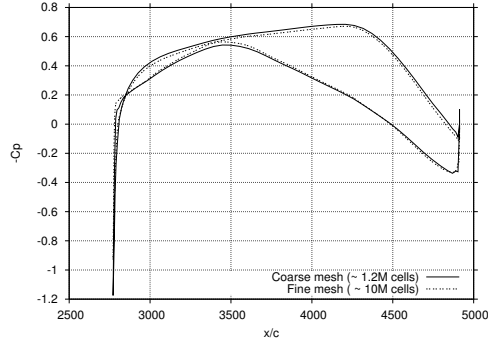
the pressure distribution which will feed the finite difference boundary layer solver and the aerodynamic coefficients to be introduced within the objective function expression.

#### 3.2.5.1 Baseline analysis

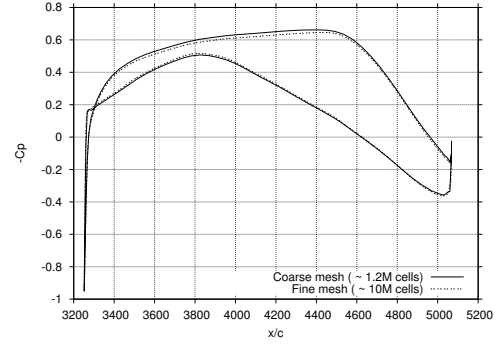
The baseline geometry is defined by mounting the optimal airfoil, designed as described in section 3.2.3, onto the wing planform: results of the aerodynamic analysis of the new baseline have been obtained both on the fine mesh and on the coarser one. The adopted turbulence model is the  $k - \omega$  TNT model with a free-stream turbulence intensity level of 0.50% and a viscosity ratio  $\mu_T/\mu$  of 0.10. CFD computation is performed at  $0^\circ$  angle of attack, which gives, on the fine mesh, a global lift coefficient of 0.24 and a drag coefficient of 220 drag counts. Pressure distribution comparisons are reported in Figure 3.25(a), Figure 3.25(b), Figure 3.25(c) and Figure 3.25(d), where red and black lines indicate respectively coarse and fine solution: the discrepancies can be considered acceptable. Figure 3.25(e) and Figure 3.25(f) compare the non-dimensional wall distance  $y^+$  at two wing sections (20% and 75% of span length) as computed with the coarse and fine mesh: even if the numbers are rather different, the unit order magnitude is preserved on the coarse mesh, as expected. Globally, the comparison between the solutions obtained with the two grid levels shows no remarkable differences. This confirms that, from the perspective of natural laminar flow design, the coarse grid level is a suitable choice to speed up the design process.

It has to be pointed out here that the baseline design is found to be already a sub-optimal candidate thanks to the careful airfoil design activity. The wing spanwise loading is compared to the elliptical distribution in Figure 3.26(a): the overall agreement is already acceptable and just a slight twist adjustment is needed on the very outboard wing to improve the low speed stall path characteristics in such a region. Figure 3.26(b) shows the predicted laminar to turbulent transition locations (the curves in black on pressure coefficient contour map) as computed on the baseline wing suction and pressure side: the laminar regions (the areas between the leading edge lines and the black curves) are very extended as the transition lines lie where the aft recompression starts, except in the inboard wing where the three-dimensional cross-flow effects are stronger and trigger an early transition to turbulence. Moreover, as expected, a negligible influence of engine mass flow is found on the outer wing aerodynamics due to its installation position. This analysis gives sense to the chosen parametrization

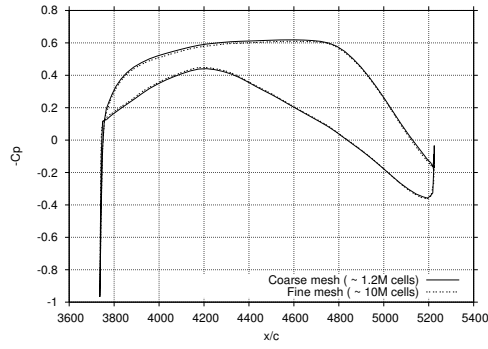
### 3.2 A state of the art optimization study



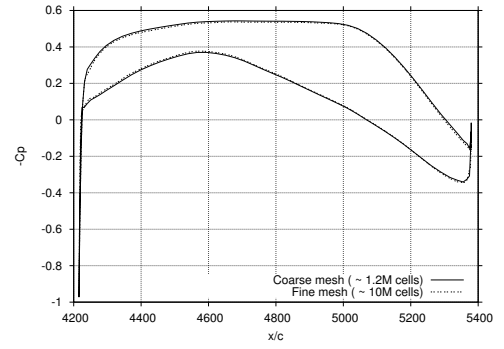
(a) Pressure distribution at 20% span



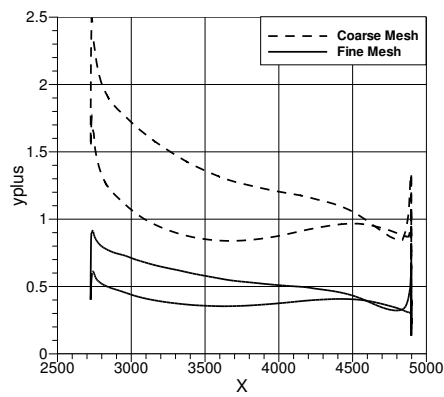
(b) Pressure distribution at 40% span



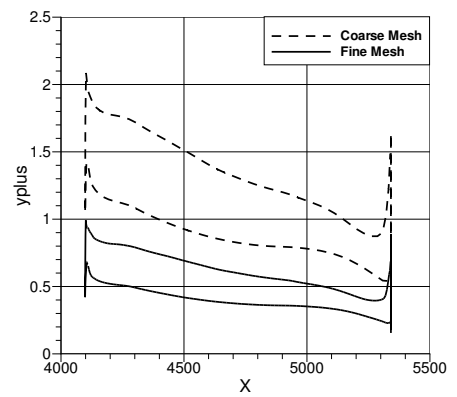
(c) Pressure distribution at 60% span



(d) Pressure distribution at 80% span



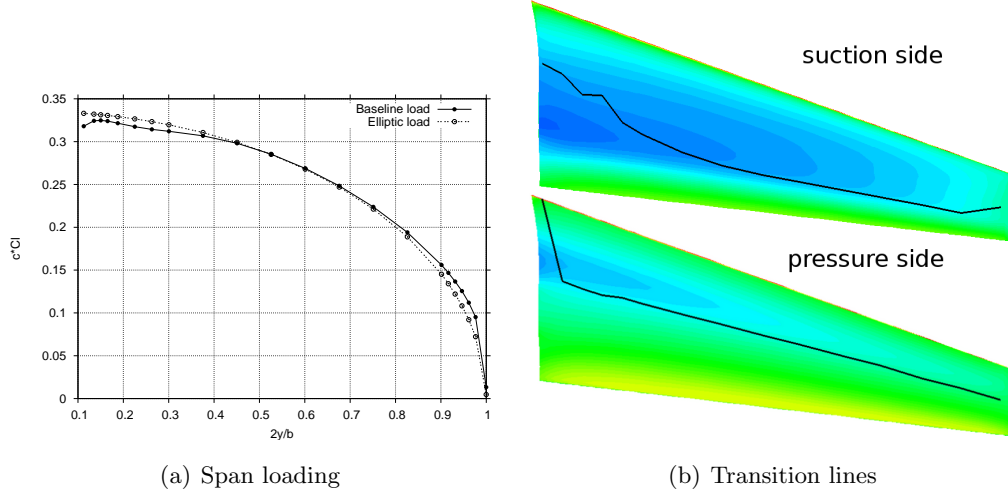
(e)  $y^+$  distribution at 20% span



(f)  $y^+$  distribution at 80% span

**Figure 3.25:** Baseline wing analysis

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION



**Figure 3.26:** Baseline wing analysis

approach: globally, the baseline wing shape can be considered as an optimal candidate, but further shape modifications are needed to improve the span loading near the wing tip (twist modification) and, possibly, the laminar extent near the wing root (inboard sectional airfoil modification).

#### 3.2.5.2 Discussion of optimization results

As highlighted in the previous section, thanks to the intensive 2.5D optimization strategy, the baseline wing, obtained by shaping the wing with the optimized airfoils, is already a good optimal or sub-optimal candidate. However, some slight improvements should be still achieved in order to correct three-dimensional effects. In particular, a reduced parameterization can be used to focus the shape modification only in the inboard wing (60 modification function variables) and the twist angle tuning on the whole wing (3 design variables, one for each shape section) to compensate the span loading. Hence, a total number of 63 design variables is used. The aircraft angle of attack is kept fixed at  $0^\circ$  throughout the optimization process and two significant features arise according to the problem definition stated in table 3.2:

- in the objective function the aerodynamic efficiency, not the drag coefficient, explicitly appears to take into account both drag minimization and lift maximization;

### 3.2 A state of the art optimization study

- the lift coefficient is quadratically penalized two times in order to express both the upper and lower bound constraints.

The resulting optimization setup for the final 3D phase is reported in table 3.3. As already mentioned, even if apparently the objectives to maximize are more than one, a single objective approach is chosen by condensating the aerodynamic efficiency (computed by the CFD solver) and the laminar extent (computed by the stability analysis) functions and selecting suitable scaling factors. Indeed, the cost function is expressed as

$$Obj = -a'' \frac{C_L}{C_D} - b'' \tilde{X}^{up} - c'' \tilde{X}^{lo} + d'' (C_L - \overline{C_L})^2 + e'' (\overline{C_L} - C_L)^2 + f'' (C_M - \overline{C_M})^2$$

where the aerodynamic efficiency  $\frac{C_L}{C_D}$ , the upper  $\tilde{X}^{up}$  and lower  $\tilde{X}^{lo}$  laminar extent are taken with the minus sign in order to set a minimization problem,  $\overline{C_L}$  and  $\overline{C_L}$  are the lower and upper bound of the  $C_L$  range (0.20 and 0.30 respectively, according to table 3.2) and  $\overline{C_M}$  is the threshold value for pitching moment coefficient ( $= -0.1$ ).  $a'', b'', c'', d'', e'', f''$  are scaling factors.

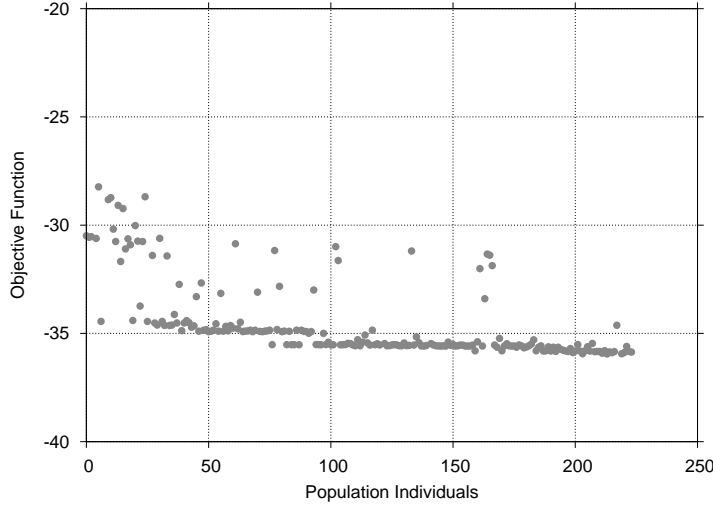
A genetic algorithm optimization is launched with a bit-mutation rate of 0.7% and a crossover rate of 80%. The need of computational effort reduction leads to a choice

**Table 3.3:** 3D Optimization problem definition

Design variables	Wing twist Inboard wing sect. shape	3 section, 3 des. var. 1 sect., 60 des. var.
Design point 1 (Cruise Priority 1)	Mach Reynolds Lref [m] Angle of attack [°] Altitude [m]	0.694 9.0E+06 1.786 0.0 35000
Constraints	Wing $C_L$ @ DP1 Wing $C_M$ @ DP1	[0.20 : 0.30] > -0.1
Objectives	$\frac{C_L}{C_D}$ @ DP1 Suction side laminar extent $\tilde{X}^{up}$ @ DP1 Pressure side laminar extent $\tilde{X}^{lo}$ @ DP1	To be maximized To be maximized To be maximized

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

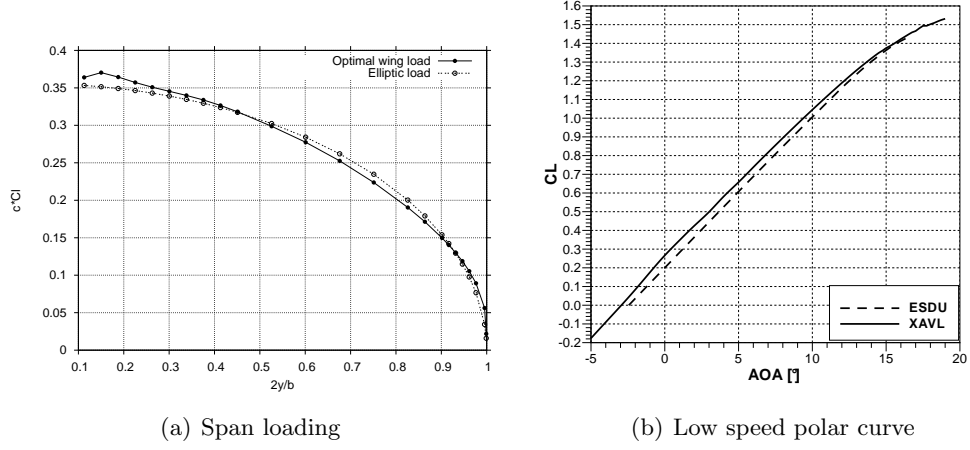
---



**Figure 3.27:** GA convergence history

for population size of only 16 elements. Based on previous experience, this is the minimum acceptable size for the used algorithm and the given optimization problem. Figure 3.27 reports the convergence history of the genetic optimization process during 14 generations. After that, the process has been stopped for two main reasons: on one hand, the best candidate performances fully satisfy the requirements; on the other hand, looking at the convergence history, it seems that a flat region, a sort of *plateau* has been already found where the solution does not improve significantly anymore. Of course, the last sentence is just a trend extrapolated from available data and, hence, it should be confirmed by further investigations. The low speed design point is not considered here as limited deviations are expected in span loading with respect to the baseline wing. Moreover, as the baseline wing is already a well-designed starting point, the designer is not interested in modifying very much the wing shape, so that the design variables range in a narrow interval and, consequently, due to the adopted reduced parameterization, they will generate very “similar” shapes.

Globally, the optimization run has achieved an improvement of the objective function of about 17% with respect to the baseline. CFD computation at  $0^\circ$  angle of attack (i.e., the design point) gives, on the fine mesh, a global lift coefficient of 0.26 and a drag coefficient of 225 drag counts. Figure 3.28(a) shows the comparison between the minimum induced drag and optimized wing span loading. The comparison with Figure 3.26(a) clarifies some of the effects of the optimization process: due to the twist angle

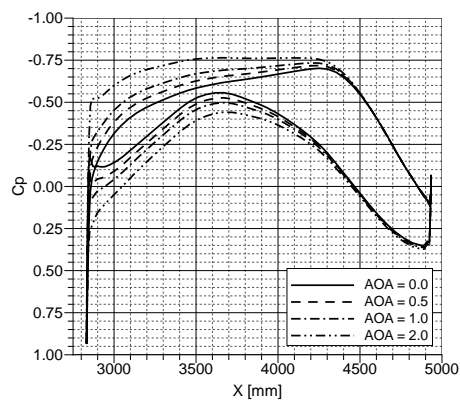


**Figure 3.28:** Optimal wing analysis

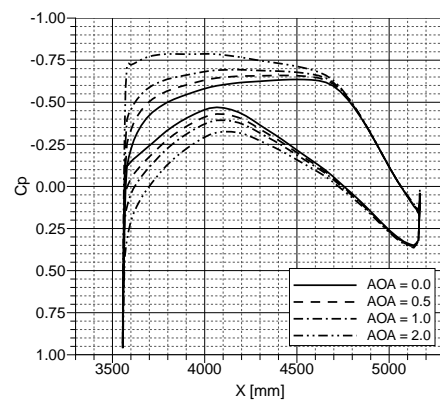
increase in the inboard wing and decrease in the mid-outboard part, the lift distribution is much more shifted near the wing root and this is a highly desirable feature both for low speed stall path improvement and structural bending moment reduction. The low speed lift polar curve is reported in Figure 3.28(b) where the results from two different methods are compared: one is the classical ESDU procedure, the other one (XAVL) is based on a vortex lattice method corrected with a strongly-coupled viscous-inviscid interaction technique. Both methods agree on maximum lift and stall angle prediction, even if they differ in the linearity region. This is surely due to the fact that the ESDU procedure was traditionally coded to give fast and quite accurate predictions in high lift conditions, while the linear lift curve slope and the zero-lift angle are roughly computed from geometry input data. However, the prediction from both methods confirm that the optimized wing satisfies the constraint on maximum lift  $CL_{max} > 1.4$ . Figure 3.29(a), Figure 3.29(b) and Figure 3.29(c) show the pressure coefficient distribution extracted respectively from 25% (inboard), 50% (mid-wing) and 75% (outboard) span sections. The results from computations at various angle of attacks ( $0^\circ$ ,  $0.5^\circ$ ,  $1^\circ$  and  $2^\circ$ ) have been superimposed on each graph. It can be observed that, on the suction side approximately up to 70% chord, the pressure gradient is constantly negative and gradually decreasing in modulus: this trend is not observed at  $AOA=2^\circ$ , where the mid-wing and outboard airfoils show an early leading edge pressure peak and a positive  $c_p$  slope downstream. This is due to the fact that, for an angle of attack of  $2^\circ$ ,

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

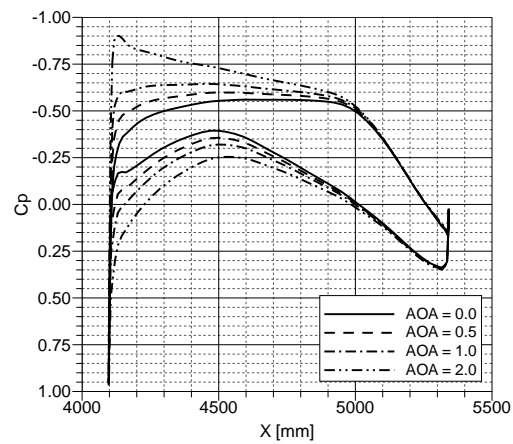
---



(a) Inboard wing section



(b) Mid wing section



(c) Outboard section

**Figure 3.29:** Optimal wing pressure distribution



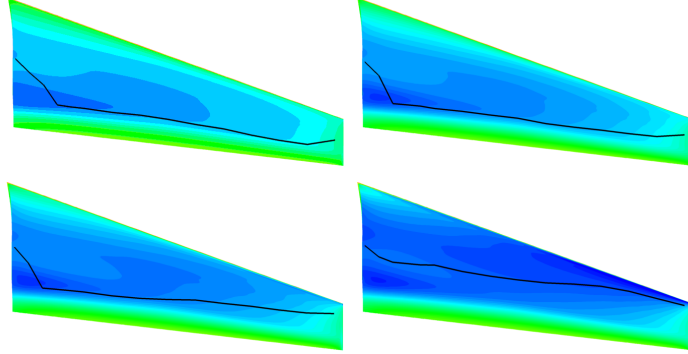
### 3.2 A state of the art optimization study

---

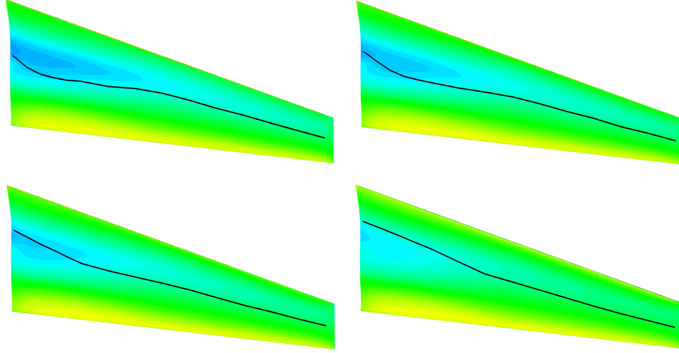
the global lift coefficient is about 0.50, which is very far from the lift design range (0.20 – 0.30): hence, the  $\text{AOA}=2^\circ$  case is a deep off-design point. On the pressure side, the pressure gradient becomes positive around the 40% chord to allow the airfoil loading increase in the aft region. This is a direct consequence of designing a wing to be the more laminar possible once a given thickness distribution and a required total lift have been assigned. The main result of the design activity is highlighted in Figure 3.30(a) and Figure 3.30(b), where the transition lines are respectively depicted on upper and lower pressure contour map for cases  $\text{AOA}=0^\circ$ ,  $0.5^\circ$ ,  $1^\circ$  and  $2^\circ$ . The  $0^\circ$  case is the upper left one on both figures: the remaining cases follow in clockwise order with  $\text{AOA}$  increasing. Transition is computed here with the database method and for a critical  $N$  factor of 15. The pictures show that a satisfying laminar performance is maintained at least until  $\text{AOA}=1^\circ$  which correspond to  $C_L = 0.38$ , that is much higher than the upper limit of the lift design range (0.30). The  $\text{AOA}=2^\circ$  case exhibits a significant loss in laminar performance as it represents a flow condition very far from the design point, as also highlighted above. These results are confirmed and better quantified in Figure 3.31(a) and Figure 3.31(b), where the percentage of laminar surface is plotted against the lift coefficient respectively on suction and pressure side. In other words, the plotted variable represents the ratio between the laminar wetted area and the wing planform area. The pictures show that a satisfying laminar performance is maintained at least until  $\text{AOA}=1^\circ$  which correspond to  $C_L = 0.38$ , that is much higher than the upper limit of the lift design range (0.30). The  $\text{AOA}=2^\circ$  case exhibits a significant loss in laminar performance as it represents a flow condition very far from the design point. Computations are reported for baseline and optimal wing (solid lines) while the circles represent the results obtained by using the fine mesh flow solution on the optimized wing. The curves give a trend of the laminar extent behavior with the wing lift level: clearly, the peak is found around  $C_L = 0.26$  which is the computed design point. However, the noticeable result is that, unlike the baseline, the optimized wing respects very well the initial requirements on laminar flow: indeed, the laminar extent is kept well higher than 60% on the suction side and 40% on the pressure side across the entire range of target lift coefficient and more. This gives further confidence in the successfulness of the chosen approach to the design process. Figure 3.32(a) and Figure 3.32(b) report the lift-drag aerodynamic polar and the aerodynamic efficiency curve as a function of  $\text{AOA}$ . The comparison allows to quantify a difference of about

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---



(a) Suction side

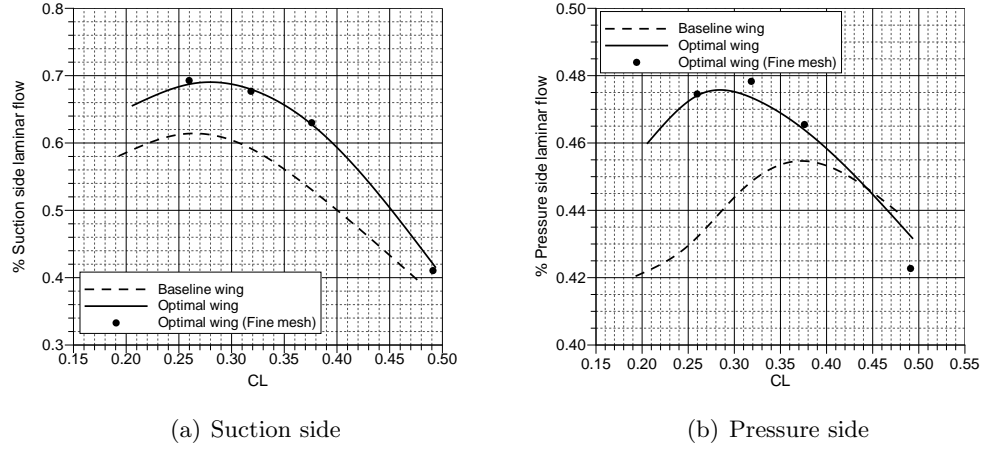


(b) Pressure side

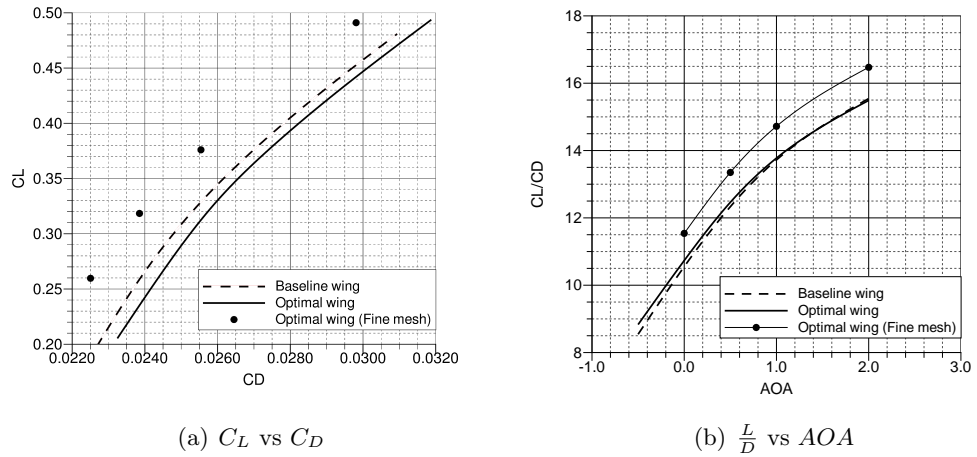
**Figure 3.30:** Transition lines on pressure contours at various AOAs

20 drag counts between the coarse and fine mesh predictions on the optimized wing. Moreover, looking at Figure 3.32(a), it might seem that the baseline wing exhibits a lower drag coefficient than the optimized one: however, it must be pointed out that, as the design process has been performed by fixing the AOA, and not the lift coefficient, the fair comparison is the one in Figure 3.32(b), where actually the optimized wing shows slightly better performances than the baseline. Indeed, near the design point and taking the same angle of attack, the optimized wing produces more lift and drag than the baseline, but the relative difference in lift overcomes the relative difference in drag, thus resulting in a higher aerodynamic efficiency. This is the reason why in the optimization process the target was put on maximizing the lift to drag ratio rather than minimizing the pure drag and the desired lift level was controlled through additional constraints.

### 3.2 A state of the art optimization study



**Figure 3.31:** % Laminar extent comparison



**Figure 3.32:** Polar curves comparison

### 3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION

---

#### 3.2.6 Concluding remarks

A state of the art optimization study has been proposed to set a baseline point for further research and to illustrate the need for speeding up the optimization process. The design of a natural laminar flow wing for a VLJ wing-body-nacelle configuration has been performed with a peculiar optimization strategy, given the complex and time-consuming problem: suitable and specific airfoils have been designed first with quasi three-dimensional aerodynamic characteristics to match wing flow conditions, the real 3D flow and the presence of fuselage and rear-mounted engine in working conditions have been simulated after to “correct” the shape of the optimal wing airfoils and its performances. Both optimization stages have shown interesting characteristics and successful results: at the end of the design process, a wing geometry has been generated with optimal aerodynamic characteristics. In particular, the objective to design a natural laminar flow wing has been completely achieved in a wide range of working conditions.

The proposed optimization example shows that a difficult and industrially challenging problem can be effectively solved using standard and well consolidated Evolutionary Computing tools. The required computational effort, although significant, is well within the limits that can be considered acceptable in an industrial development environment. Nevertheless, that particular case (low-sweep wing at low transonic Mach numbers) allowed us to adopt a peculiar strategy to avoid a prohibitive computational load on the full three-dimensional configuration: a different study case could not be treatable in that way, forcing the designer to find a more general way to reach the target in a reasonable time and with limited resources. In this context, the search of more efficient optimization techniques is still very desirable. Higher efficiency generally means:

- using lower-fidelity or lower-dimensional analysis tools (e.g., Euler instead of Navier Stokes flow model, two-dimensional approximation instead of three-dimensional as shown above), possibly combined with the high-fidelity tool in a multi-fidelity environment;
- improving the search algorithm through hybridization (e.g., gradient methods combined with genetic optimization);

### **3.2 A state of the art optimization study**

---

- using highly-accurate, physics-based surrogate models to avoid to turn to or to limit the expensive CFD computation during the iterative process.

The following chapters will be devoted to the study and development of a methodology belonging to the third class with particular focus on transonic aerodynamics and shock wave treatment in shape optimization problems.

### **3. INGREDIENTS FOR AERODYNAMIC DESIGN OPTIMIZATION**

## 4

# The Proper Orthogonal Decomposition

In this chapter a review of the mathematical theory of proper orthogonal decomposition is presented. As already described in previous sections, the proper orthogonal decomposition is a mathematical procedure that allows to perform a modal decomposition of large set of multi-dimensional data so as to derive a dimensionality reduction and describe the original system with much less number of unknowns. The mathematical development of POD for fluid flow applications, in particular, is described in some detail in (21). Here, the main aspects related to the construction of a reduced order model through singular value decomposition are presented and the use of this technique for steady state problems is mainly addressed. It must be underlined that, even if the POD is mostly applied to non-linear problems because their solution is usually expensive, it remains a linear procedure as the subspaces generated by POD modes are linear spaces.

Note that although the POD methodology is almost exclusively applied to non-linear problems, it is important that one recognizes that it is a linear procedure, and the nested sequence of subspaces are linear spaces, even if the data that generates it are non-linear.

### 4.1 Reduced order models

Physics-based approximation concepts require a deep understanding of the governing equations and the numerical methods employed for their solution. One of the notable

#### 4. THE PROPER ORTHOGONAL DECOMPOSITION

---

advantage of most physics-based approaches is that they are quite insensible to the curse of dimensionality, hence they do not show any limitations when the number of design variables notably increases. This is one of the most important advantages of such approximations with respect, for example, to Kriging.

By using reduced basis methods it is possible to build models with fewer unknowns than the original high-fidelity model. To illustrate reduced basis methods consider the discrete mathematical model (e.g., RANS equations) of a physical system written in the form

$$\mathbf{R}(\mathbf{w}, \mathbf{x}(\mathbf{w})) = 0 \quad (4.1)$$

where  $w \in \mathbb{R}^p$  is the vector of design variables and  $\mathbf{x} \in \mathbb{R}^q$  the discretized vector of state (or field) variables (velocity, energy, density). Note that  $\mathbf{x}$  is an implicit function of the design variables. Classical interpolation/approximation methods (e.g., Kriging, RBF) do not work on equation 4.1 and the relation between state and design variables, but they look just to local or integral values of the state variables. The power of reduced order methods, instead, lies in the fact that they give an approximation of the state vector in the form

$$\hat{\mathbf{x}} = c_1\phi_1 + \dots + c_M\phi_M = \mathbf{\Phi}\mathbf{c} \quad (4.2)$$

where  $\mathbf{\Phi} = \{\phi_1, \dots, \phi_M\} \in \mathbb{R}^{q \times M}$  is a matrix of known basis vectors and

$$\mathbf{c} = \{c_1, \dots, c_M\} \in \mathbb{R}^M$$

is a vector of unknown coefficients. This kind of approximation supposes that any response vector belongs to the subspace spanned by the chosen basis vectors. This in general depends on how the basis vectors have been computed. Hence, the original problem with  $q$  unknowns can be recast into a problem with  $M$  unknowns and so if  $M \ll q$  then it is possible to approximate  $x$  very efficiently. Different methods exist to estimate the approximation coefficients, generally classified in two classes: those using the governing equations and minimizing some residual norm and those employing accurate data-fitting techniques starting from a set of known coefficients.

A good set of basis vectors is a set of vectors which is easy to compute as well as guaranteed to be linearly independent. The general, widely used choice to select the basis vectors is to use state solutions of the discrete governing equations which



correspond to  $M$  different values of the parameters  $w$  and compute  $\Phi$  as a basis of the subspace:

$$\Phi = \text{span}\{\mathbf{x}(\mathbf{w}_1), \mathbf{x}(\mathbf{w}_2), \dots, \mathbf{x}(\mathbf{w}_M)\} \in \mathbb{R}^{q \times M} \quad (4.3)$$

For example, these solutions are obtained by solving the RANS equations, which give the state vector  $x$ , on  $M$  different configurations generated by applying the employed parameterization method on  $M$  design vectors  $\mathbf{w}_i$ . The definition of the  $M$  design sites where to compute the solutions is not a trivial issue: generally speaking, standard design of experiments techniques are used to sample the design space with good coverage properties, but, as we will discuss in next chapters, this approach may lead to erroneous results when we face highly multi-modal, highly non-linear problems. Indeed, the quality of the approximation strongly depends on the location of training data in the design space. A small point of concern with the solution subspace is the possibility that some of the basis vectors could be linearly dependent but this can be overcome for example by employing the SVD scheme.

It is possible to apply linear reduced basis methods to non-linear problems provided a suitably rich set of basis vectors is available. In this thesis, an approach that uses the proper orthogonal decomposition method to derive the basis vectors is considered. This methodology has already been introduced in chapter 1 and a further detailed mathematical description is presented in the next section.

## 4.2 POD theory

Let  $\{\mathbf{u}_j(z)\}$  be an ensemble of real vector fields on the domain  $z \in \Omega$ . Here, it is assumed that the ensemble consists of a set of instantaneous snapshots of a numerical simulation solution field. In seeking good representations of members of  $\{\mathbf{u}_j(z)\}$ , it is required to project each  $u$  onto candidate basis functions and so it is assumed that the  $u$  s belong to an inner product space: the linear, infinite-dimensionale Hilber space  $H(\Omega)$  of square integrable functions with inner product

$$(\mathbf{f}, \mathbf{g}) = \int_{\Omega} f(z)g(z)dz \quad (4.4)$$

The POD basis is a set of functions  $\{\phi_j(z)\}$  that is optimal in describing the ensemble of observations. Indeed, they are found by maximizing the averaged projection of the ensemble  $\{\mathbf{u}_j\}$  onto  $\phi$  :

#### 4. THE PROPER ORTHOGONAL DECOMPOSITION

---

$$\max_{\phi \in H(\Omega)} \frac{\langle (\mathbf{u}, \phi)^2 \rangle}{(\phi, \phi)} \quad (4.5)$$

The averaging operator  $\langle \rangle$  used in equation 4.5 could be an ensemble average over many numerical or experimental realizations, or it could be a time-average taken from different samples of a single experiment. The main assumption regarding the averaging operator is that it commutes with the inner product. This assumption is true for the scalar case defined on the Hilbert space  $L^2$  under certain conditions on  $u$ .

Since the basis is linear, a flow-field  $\mathbf{u} \in \text{span}\{\phi_j\}$  can be represented as a linear combination of the POD modes

$$\mathbf{u}(z, w) = \sum_i \alpha_i(w) \phi_i(z) \quad (4.6)$$

where the dependency on the design site  $w$  has been introduced to underline the empirical character of the ensemble. If we assume that  $(\phi, \phi) = 1$ , i.e. the basis vectors are normalized, the constrained optimization problem 4.6 reduces to the eigenvalue problem

$$\mathbf{R}\phi = \lambda\phi \quad (4.7)$$

where

$$\mathbf{R} = \langle \mathbf{u}_j(\mathbf{u}_j, \phi) \rangle \quad (4.8)$$

The operator  $\mathbf{R}$  is self-adjoint and non-negative definite; by also assuming that it is a compact operator, then there exists a countable set of non-negative eigenvalues  $\lambda_i$  with associated eigenfunctions  $\phi_i$ . The eigenfunctions, once normalized, form an orthonormal subspace of  $H$ , i.e.  $(\phi_i, \phi_j) = \delta_{ij}$ . For more details about the compactness of  $\mathbf{R}$  and the required assumptions, refer to Lumley et. al ().

The POD modes are the eigenfunctions  $\phi_i$  associated with non-zero  $\lambda_i$ . Taking the inner product of equation 4.7 with  $\phi$ , it is straightforward to show that  $\langle (\mathbf{u}_j, \phi_i)^2 \rangle = \lambda_i$ . In other words, the magnitude of the eigenvalue is equivalent to the average energy of the projection of the ensemble onto the associated eigenfunction, where the square of the inner product is interpreted as an energy measure. It is important to remark that in incompressible fluid mechanics with velocity measurements, this energy is related to the fluids kinetic energy. The POD modes may be ordered according to the magnitude of

their eigenvalue, with  $\lambda_1/\phi_1$  equal to the eigenvalue/eigenfunction pair with the largest eigenvalue,  $\lambda_M$  equal to the smallest non-zero eigenvalue, and  $\lambda_1 \geq \lambda_2 \dots \geq \lambda_M$ . In building reduced order models one is interested in truncating the POD basis and retaining only the most energetic modes. It can be shown that the sequence of truncated POD bases form an optimal set, in the sense that a POD basis comprised of  $K$  modes describes more energy (on average) of the ensemble than any other linear basis of the same dimension  $K$ . This compression of the ensemble energy into a minimum number of modes makes the POD basis attractive for reduced order modelling. The span of the POD basis is not complete in  $H(\Omega)$ , but it is complete in the sense that, on average, any snapshot used to construct it can be represented, i.e.

$$\langle \|\mathbf{u}_j - \sum_i (\mathbf{u}_j, \phi_i) \phi_i\| \rangle = 0 \quad (4.9)$$

Conversely, each POD mode can be obtained as a linear combination of the observations used to construct the basis.

In practice, the  $\mathbf{u}_j$  are vectors of state variables at discrete grid point locations, each containing a single solution from the numerical simulation. They will have length  $v \times q$ , where  $q$  is the total number of grid points and  $v$  is the number of dependent variables describing the flow state. Thus, the discretized version of eqn. 4.7 will be an eigenvalue problem of order  $v \times q$ . For  $q \gg M$ , where  $M$  is the number of flow field snapshots used, this procedure is costly and inefficient.

Sirovich () showed how the eigenvalue problem 4.7 can be reduced to order  $M$ , resulting in a much more efficient procedure for  $q \gg M$ . Equation 4.7 can be rewritten by introducing the modal decomposition of the ensemble and supposing the averaging operator as an ensemble average

$$\frac{1}{M} \sum_{j=1}^M \mathbf{u}_j (\mathbf{u}_j, \phi_i) = \frac{1}{M} \sum_{j=1}^M \mathbf{u}_j \alpha_i^{(j)} = \lambda_i \phi_i \quad (4.10)$$

where the  $i$ -th modal coefficient of the  $m$ -th snapshot, generated at design site  $w_m$ , is defined as

$$\alpha_i(w_m) = (\mathbf{u}_m, \phi_i) = \sum_{k=1}^{v \times q} u(z_k, w_m) \phi_i(z_k) \quad (4.11)$$

## 4. THE PROPER ORTHOGONAL DECOMPOSITION

---

From equation 4.10, each component  $\phi_i(z_k)$  of the basis vectors can be written as

$$\phi_i(z_k) = \frac{1}{\lambda_i} \frac{1}{M} \sum_{j=1}^M u(z_k, w_j) \alpha_i(w_j) \quad (4.12)$$

Substituting this expression back into equation 4.11 and changing the order of summation gives

$$\sum_{j=1}^M \frac{1}{M} \sum_{k=1}^{v \times q} u(z_k, w_m) u(z_k, w_j) \alpha_i(w_j) = \lambda_i \alpha_i(w_m) \quad (4.13)$$

In a more compact form, equation 4.13 reads as

$$\sum_{j=1}^M C_{mj} \alpha_i(w_j) = \lambda_i \alpha_i(w_m) \quad (4.14)$$

This has to be solved for all the modes, so the resulting system is made of  $M$  equations for the eigenvalues  $\lambda_i$  and coefficients  $\alpha_i(w_j)$ . It is now clear why the method of snapshots allows to easily find the empirical eigenfunctions when  $q \gg M$ . The POD modes can be recovered by using equation 4.10, which states that the spatial eigenfunctions are weighted sums of the original set of realizations or snapshots  $\{\mathbf{u}_j\}$ .

### 4.3 POD solution

In this section, the POD is described for steady-state problems in terms of the singular value decomposition (SVD). This approach is normally preferred as it is more straightforward and easy to be implemented.

#### 4.3.1 Snapshots collection

For each design site  $w$ , a CFD solution is computed with the flow solver and the flow variables are stored to form a vector  $S \in \mathbb{R}^N$  called snapshot:

$$\begin{aligned} S &= [S_{grid}, S_{flow}]^T \\ S_{grid} &= [x_1, \dots, x_q, y_1, \dots, y_q, z_1, \dots, z_q] \\ S_{flow} &= [\rho_1, \dots, \rho_q, \rho u_1, \dots, \rho u_q, \rho v_1, \dots, \rho v_q, \rho w_1, \dots, \rho w_q, p_1, \dots, p_q] \end{aligned}$$

where  $q$  is the number of mesh nodes involved in the POD computation,  $(x, y, z)$  are the nodes coordinates,  $\rho$  is the flow density,  $(u, v, w)$  are the three velocity components and

$p$  is the static pressure. The global size of the snapshot is  $N = 8 \times q$ . Each snapshot is constructed by placing in order the solution at each grid point for the whole grid. This order can be determined arbitrarily, but is subject to the constraint that it must be consistent throughout the whole set of snapshots. Moreover, the POD modes are sensitive to the scaling of the flow variables and, as the dataset is made of heterogeneous variables as these are in different units and have significantly varying magnitudes, this can represent a big issue. Consequently, appropriate scaling factors have to be applied for each fluctuating flow variable in order to get an almost uniform snapshot. A set of snapshots  $S_j, j = 1, \dots, M$  computed at design sites  $w_j$  is called an ensemble, where usually  $M \ll N$ . The set of design sites is usually obtained by means of a Design of Experiments technique which samples the design space.

#### 4.3.2 SVD solution

Starting from the vectors  $S_1, S_2, \dots, S_M$  obtained by the CFD expensive computations for a representative set of design sites  $w_1, w_2, \dots, w_M$ , finding a Proper Orthogonal Decomposition means to compute a linear basis of vectors to express any other  $S_j \in \mathbb{R}^N$  with the condition that this basis is optimal in some sense. To compute the optimal basis, we first define the snapshot deviation matrix

$$P = [S_1 - \bar{S}, S_2 - \bar{S}, \dots, S_M - \bar{S}] \in \mathbb{R}^{N \times M}$$

where the ensemble mean vector is computed as

$$\bar{S} = \frac{1}{M} \sum_{j=1}^M S_j \in \mathbb{R}^N$$

Then, we search for a set of orthonormal vectors  $\phi_1, \phi_2, \dots, \phi_M$  such that

$$S_j = \bar{S} + \sum_{i=1}^M \alpha_i^j \phi_i = \bar{S} + \sum_{i=1}^{\hat{M}} \alpha_i^j \phi_i + \epsilon_{\hat{M}}^j = \bar{S} + \sum_{i=1}^{\hat{M}} (S_j^T, \phi_i) \phi_i + \epsilon_{\hat{M}}^j$$

with  $\hat{M} \leq M$  and the error  $\epsilon$  is the smallest possible. Indeed, for any set of orthonormal vectors  $\psi_1, \psi_2, \dots, \psi_M$  the optimality condition is expressed as:

$$\epsilon_{\hat{M}} = \sum_{j=1}^M |S_j - \bar{S} - \sum_{i=1}^{\hat{M}} (S_j^T, \phi_i) \phi_i| \leq \sum_{j=1}^M |S_j - \bar{S} - \sum_{i=1}^{\hat{M}} (S_j^T, \psi_i) \psi_i|$$

#### 4. THE PROPER ORTHOGONAL DECOMPOSITION

---

Therefore, we take the singular value decomposition (SVD) of  $P$

$$P = U\Sigma V^T = U \begin{Bmatrix} \sigma_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma_M \\ 0 & \cdots & 0 \end{Bmatrix} V^T \quad (4.15)$$

with  $U \in \mathbb{R}^{N \times N}$ ,  $V \in \mathbb{R}^{M \times M}$ ,  $\Sigma \in \mathbb{R}^{N \times M}$  and the singular values  $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_M \geq 0$ . The POD basis vectors, also called POD modes, are the first  $M$  column vectors of the matrix  $U$ , while the POD coefficients  $\alpha_i^j$  are obtained by projecting the snapshots onto the POD modes:

$$\alpha_i^j = ((S_j - \bar{S})^T, \phi_i) \quad (4.16)$$

If a problem is represented by a suitable number of snapshots from which a suitably rich set of basis vectors is available, the singular values become small rapidly and a small number of basis vectors are adequate to reconstruct and approximate the snapshots as they preserve the most significant ensemble energy contribution. In this way, POD provides an efficient means of capturing the dominant features of a multi-degree of freedom system and representing it to the desired precision by using the relevant set of modes, thus reducing the order of the system. In other words, the reduced-order model is derived by projecting the CFD model onto a reduced space spanned by only some of the proper orthogonal modes or POD eigenfunctions. The following approximation is done

$$S_j \simeq \bar{S} + \sum_{i=1}^{\hat{M}} \alpha_i^j \phi_i \quad (4.17)$$

where

$$\hat{M} \leq M \implies \frac{\sum_{i=1}^{\hat{M}} \sigma_i^2}{\sum_{i=1}^M \sigma_i^2} \geq \epsilon \quad (4.18)$$

and  $\epsilon$  is a pre-defined energy level. In fact, the truncated singular values fulfill the relation

$$\sum_{i=\hat{M}+1}^M \sigma_i^2 = \epsilon_{\hat{M}}$$

If the energy threshold is high say over 99% of the total energy, then  $\hat{M}$  modes are adequate to capture the principal features and approximately reconstruct the dataset. Thus, a reduced subspace is formed which is only spanned by  $\hat{M}$  modes.

### 4.3.3 Pseudo-continuous global representation

The use of reduced-order models based upon the proper orthogonal decomposition for prediction requires the transformation of the projection coefficients  $\alpha_i^j$  from the discrete sample space for which they have been computed to a continuous space. Equation 4.17 allows to get a POD approximation of any snapshot  $S_j$  belonging to the ensemble set computed at DOE design sites  $W_j$ . Indeed, the model does not provide a POD solution in a design site which is not included in the original training ensemble. In other words, the POD model by itself does not have a global predictive feature over the full design space. To this aim, it is needed to establish a functional relation between the  $\alpha_i$  coefficients, which represent the projection of a generic CFD field onto the set of POD basis vectors, and the design variables. While regression techniques work well for fitting experimental data, where noise due to random errors is smoothed out from the data, they are less appropriate when dealing with the results from deterministic numerical simulations or when working with complex data sets. Of particular concern in the context of the present work are two problems; firstly that the response surface constructed using regression analysis may not exactly fit the sample data from which it has been constructed and secondly that the method smooths local variations in the data. In order to overcome these well-known problems, we adopt a Radial Basis Function (RBF) network over the POD coefficients to determine their values at a generic design site which is not included in the original ensemble (12). In particular,  $\hat{M}$  RBF interpolations are built on the correspondence

$$(w_1, w_2, \dots, w_M) \in \mathfrak{R}^M \rightarrow \begin{cases} (\alpha_1^1, \alpha_1^2, \dots, \alpha_1^M) \in \mathfrak{R}^M \\ (\alpha_2^1, \alpha_2^2, \dots, \alpha_2^M) \in \mathfrak{R}^M \\ \dots \\ (\alpha_{\hat{M}}^1, \alpha_{\hat{M}}^2, \dots, \alpha_{\hat{M}}^M) \in \mathfrak{R}^M \end{cases} \quad (4.19)$$

A Radial Basis Function is a real-valued function whose value depends on the Euclidean distance from a point called centre. A RBF network uses a linear combination of radial functions. Given the correspondence in 4.19, a RBF model can be expressed as

$$\alpha(w) = \sum_{i=1}^M f_i r(|w - w_i|, \theta_i) \quad (4.20)$$

where the approximating function is represented by a sum of  $M$  RBFs  $r$ , each associated with a different center  $w_i$ , weighted by a real-valued weight  $f_i$  and characterized by a

#### 4. THE PROPER ORTHOGONAL DECOMPOSITION

---

width parameter  $\theta_i$ . Hence, an RBF network can be defined as a weighted sum of translations of radially symmetric basis function. Typical RBFs  $r$  are:

- Gaussian  $r(d, \theta) = e^{-\frac{d^2}{2\theta^2}}$
- Multi-quadric  $r(d, \theta) = \sqrt{1 + \frac{d^2}{\theta^2}}$
- Inverse quadratic  $r(d, \theta) = \frac{1}{1 + \frac{d^2}{\theta^2}}$

Once fixed the types of RBF to be used and the “optimal” width parameters, the RBF network is defined only by the weights  $f_i$ . They are found by imposing the interpolation condition on the training set for any modal coefficient  $i \leq \hat{M}$

$$\begin{cases} \alpha_i(w_1) = \alpha_i^1 \\ \alpha_i(w_2) = \alpha_i^2 \\ \dots \\ \alpha_i(w_M) = \alpha_i^M \end{cases} \quad (4.21)$$

which in turn results in solving  $\hat{M}$  linear systems

$$\begin{Bmatrix} r(0, \theta_1) & \dots & r(|w_1 - w_M|, \theta_M) \\ r(|w_2 - w_1|, \theta_1) & \dots & r(|w_2 - w_M|, \theta_M) \\ \vdots & \vdots & \vdots \\ r(|w_M - w_1|, \theta_1) & \dots & r(0, \theta_M) \end{Bmatrix} \begin{Bmatrix} f_1 \\ f_2 \\ \vdots \\ f_M \end{Bmatrix} = \begin{Bmatrix} \alpha_i^1 \\ \alpha_i^2 \\ \vdots \\ \alpha_i^M \end{Bmatrix} \quad (4.22)$$

The width parameters have a big influence both on the accuracy of the RBF model and on the conditioning of the solution matrix. In particular, it has been found (12, 56) that interpolation errors become high for very small and very large values of  $\theta$ , while the condition number of the coefficient matrix increases with increasing values of  $\theta$ . Therefore, they have to be “optimal” in the sense that a tuning of the width parameters is needed to find the right trade-off between interpolation errors and solution stability (see (56) for a discussion about how to properly select the best set of parameters). The pseudo-continuous prediction of the flow field at a generic design site  $w$  is then expressed as:

$$S(w) = \bar{S} + \sum_{i=1}^{\hat{M}} \alpha_i(w) \phi_i \quad (4.23)$$

This provides a useful surrogate model which combines design of experiments for sampling, CFD for training, POD for model reduction and RBF network for global approximation. In conclusion, the model has been derived going through the following steps:



- compute a series of CFD simulations in parallel with the full order CFD model (FOM) for a representative set of design sites  $(w_1, w_2, \dots, w_M)$  in order to produce  $M$  snapshots  $S_1, S_2, \dots, S_M$ ;
- evaluate the set of POD modes  $\phi_1, \phi_2, \dots, \phi_M$  and POD coefficients as described in section 4.3.2 and according to equations 4.15 and 4.16;
- store  $\hat{M}$  modes according to equations 4.17 and 4.18;
- train  $\hat{M}$  RBF models for  $\hat{M}$  POD coefficients on  $M$  data set according to equations 4.20, 4.21 and 4.22;
- assemble the CFD field response  $S$  for an arbitrary design site as in equation 4.23.

#### 4. THE PROPER ORTHOGONAL DECOMPOSITION

---

## 5

# POD-based Reduced Order Modelling for transonic shape optimization

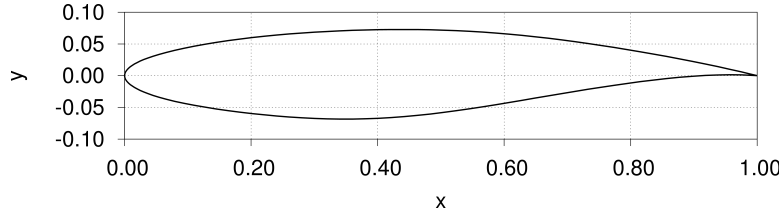
In this chapter the model described in the previous one is used to assess POD-based reduced order models (POD/ROM) for transonic aerodynamic problems. This is demonstrated by considering a viscous steady flow about a scaled RAE2822 airfoil. This case was selected as it is a standard geometry used to validate CFD numerical modelling (57). The POD snapshots are obtained by perturbing the RAE2822 airfoil by means of the parameterization described in chapter 3. A mixed POD/CFD approach (zonal POD) and adaptive sampling techniques are proposed to reach standard high-fidelity accuracy levels in shape optimization problems with limited computational cost.

## 5.1 Design space definition

The proposed methodology is aimed at providing a fast and accurate model to be used as surrogate in advanced aircraft shape design. As a consequence, one of the most important issue is to show its suitability and applicability to the shape optimization problem. Indeed, the definition of the design space through shape modification parameters typically involves a more complex, often highly non linear relation between the flow field and the design variables. Moreover, the global shape modification of an aircraft component, like a wing airfoil, requires several parameters, thus enlarging the

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

dimensions of the design space. It is straightforward, then, that the complexity of the problem under analysis increases and approaches a real-world application level. Here, a 16-dimensional design space is explored by introducing 8 design variables controlling the upper airfoil surface and 8 controlling the lower surface. Shape modifications are carried out by using the CST approach already described in chapter 3. In the present context, 7<sup>th</sup>-order Bernstein polynomials are considered, hence each airfoil side (upper and lower) is described by 8 design variables. A scaled 14% thickness ratio RAE 2822 airfoil is selected as baseline airfoil. The original RAE 2822 airfoil presents a 12% thickness ratio, but here it has been scaled to enforce stronger local transonic conditions on the airfoil surfaces. The airfoil geometry is shown in figure 5.1. The corresponding design weights, which define the RAE 2822 profile according to equations 3.8, 3.9 and 3.9, are reported in table 5.1.



**Figure 5.1:** Baseline geometry, RAE 2822 airfoil

Design weights	$\sqrt{\frac{2R_{te}}{c}}$	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$	$\tan(\beta)$
Upper side	0.1293	0.1282	0.1771	0.1219	0.2393	0.1662	0.1976	0.2110
Lower side	-0.1280	-0.1483	-0.1080	-0.2580	-0.0918	-0.1079	-0.0561	0.0638

**Table 5.1:** RAE 2822, 7<sup>th</sup>-order Bernstein polynomials design weights

The design space is defined as follows: the values of the baseline weights on the upper (respectively lower) side are taken as lower (respectively upper) boundary of the allowable variation ranges of the design weights. The global width of the range of variation is 0.1. This means, for example, that the first design weight on the upper airfoil side is allowed to vary from 0.1293 to 0.2293, while the first design weight on the lower side is allowed to vary from -0.2280 to -0.1280. In the following, every combination of the 16 design weights will be called a design site and referred to as  $w$ .

## 5.2 POD/ROM with varying geometry and grid

As the POD snapshots are generated on different geometries and, hence, they include different computational meshes, a first issue to be discussed is how to treat this discrepancy. When the physical domain between snapshots is different, that is there exists a geometrical change between snapshots, the modes derived from the snapshots will no longer remain at fixed places within the computational domain and consequently an error is introduced in this modelling technique being a space-index transformation. One approach to this problem is to use a common domain for every snapshot and apply transpiration boundary conditions to account for the changes in the boundary (58). Another approach was suggested and adopted by LeGresley and Alonso (), consisting in including the cell volume into the scalar product expression in order to take into account the mesh change. Here, quite a different approach is used. In section 4.3.1 the snapshot  $S$  structure has been presented as a combination of mesh coordinates ( $S_{grid}$ ) and flow field variables ( $S_{flow}$ ). This represents an innovative choice, because POD fields are usually provided and predicted on fixed meshes. Indeed, as the presented methodology is mainly aimed at shape design and optimization, this assumption is hardly satisfied, except for introducing complex interpolations to reduce the changing CFD fields to a fixed underlying grid. The idea proposed here is to provide the snapshot with both the flow fields variables and the mesh coordinates in order to let the POD basis catch the coupling effects between them. Hence, once the surrogate model is built, not only a flow field can be computed, but also an approximation of the volume mesh. This approach will provide a surrogate model which is already able to take into account, although in a model reduction form, the cross effects of surface geometry modification and aerodynamic flow change.

## 5.3 Design of Experiments

In the description of the surrogate/ROM modelling techniques approached so far, it has been assumed that a set of observational data was generated by some technique and is already available. In the development of any surrogate model, the location of sample points within the parametric space has an important influence on both the cost of constructing the model and on the accuracy of model predictions. In this section, the problem of generating training data that leads to approximation models that generalize

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

well is examined. Design-of-Experiment theory (DoE) (59) is a branch of statistics which provides the statistician with methods for selecting the independent variable values for a limited number of experiments, while achieving a good representation over the parametric space of interest. The various experimental design methods create certain combinations of analyses in which the independent variables are prescribed at specific values or levels. The results of these planned experiments are then used to study and investigate the response and sensitivity of some dependent quantity to the independent variables. Classical DoE techniques include full- and fractional-factorial designs. A common feature of these techniques is that the sample points are placed at the extremes of the parameter space to alleviate the effects of noise and so these are appropriate if the data is contaminated by noise, which make it necessary to employ regression techniques to filter noise. In contrast to physical experiments, observations made using computer experiments are not subject to random errors and so to extract the maximum information about the input-output relationship, the sample points are chosen to fill the design space in an optimal sense. In this work we have adopted the classical Latin Hypercube sampling (LHS) technique to select initial points for building surrogate/ROM models.

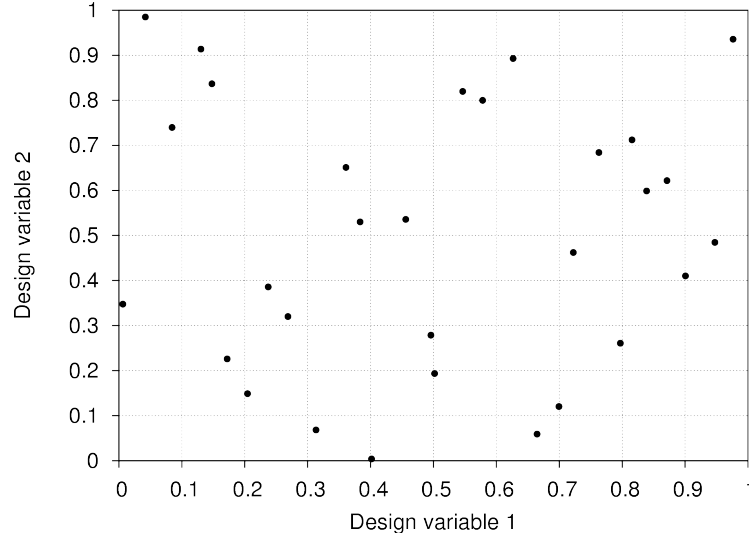
### 5.3.1 Latin Hypercube sampling

The Latin Hypercube sampling (LHC) technique was first proposed by McKay et al. (60) as an alternative to Monte Carlo techniques for the design of computer experiments. In LHS, each parameter range is divided into  $m$  intervals or bins of equal probability. If  $p$  is the number of design variables, this leads to a total of  $m \times p$  bins in the whole space. Subsequently,  $m$  samples are generated such that for each parameter, when a one-dimensional projection is taken, there will be only one sample in each bin. The LHS algorithm produces samples as follows:

$$w_j^{(i)} = \frac{\pi_j^{(i)} + \kappa_j^{(i)}}{m}, \quad \forall i, j, i = 1, \dots, m, j = 1, \dots, p \quad (5.1)$$

where  $m$  is the number of samples,  $\kappa \in [0, 1]$  is a random number and  $\pi$  is an independent random number permutation. The subscript denotes the parameter number and the superscript in brackets denotes the sample number. From each parameter, one of the points on the interval is selected randomly and the response is evaluated. This is

done until all points are used up. This method is useful because there is no correlation between parameters and the samples are chosen randomly. Figure 5.2 shows an example of LHS distribution over a two variables design space.



**Figure 5.2:** Standard LHS distribution of points

## 5.4 Zonal POD

The POD surrogate model is designed and intended to use as a reduced order model (ROM) within a shape optimization process where typically the geometry and, hence, the volume mesh around it may vary with the design site. Moreover, the application is focused on transonic aerodynamics with potential flow separations and shock waves. In other words, the target applications are close or beyond the typical aeronautical design conditions, covering almost the whole flight envelope. Care must be taken about the spatial domain where the snapshots are defined and how integral quantities of interest (e.g., aerodynamic force coefficients) or surface distributions have been computed. Indeed, due to the intrinsic linear character of the POD model, the presence of shock waves and flow separations in the initial ensemble could cause severe issues in the correct reconstruction of flow fields at unknown design sites. A zonal POD approach, similar to a domain decomposition method, is presented to get rid of the problems arising from the inclusion of flow discontinuities and intense variability sources in the POD ensemble. The basic

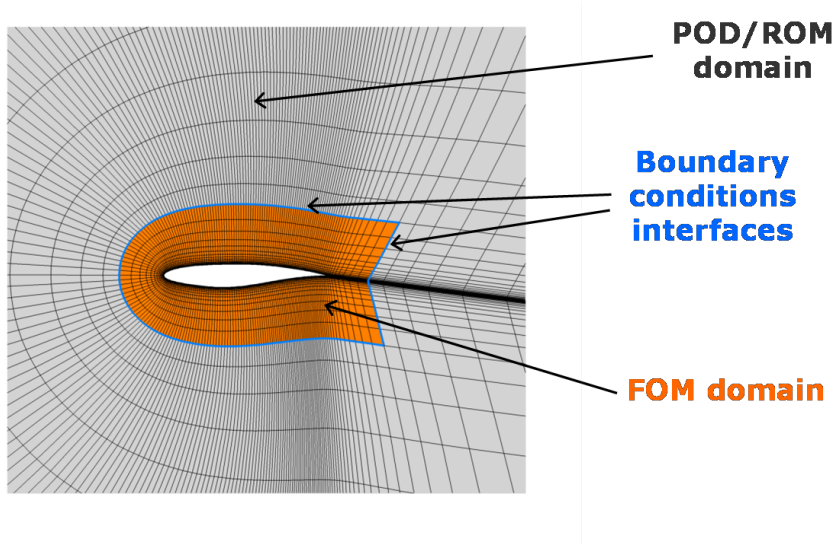
## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

idea, proposed in (60), is to use a mixed full order (FOM)/reduced order (ROM) model by splitting the solution domain into two sub-domains:

1. the FOM (i.e. the CFD RANS model) is used just around the airfoil to accurately solve the near wall boundary layer, non-linearities (e.g., shock waves) and flow separations where they occur;
2. the ROM (i.e. the POD surrogate model) is exploited to reconstruct the flow field far from the solid wall, where a smoother and weakly varying solution is expected.

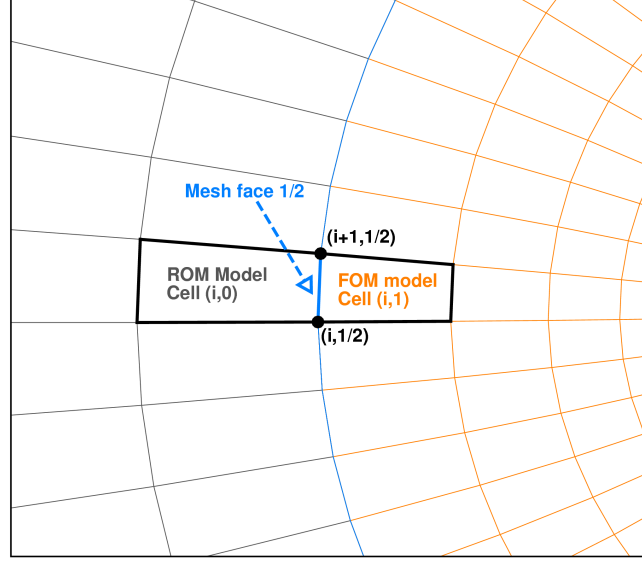
Figure 5.3 shows a sketch of the domain decomposition. The POD based surrogate model is built on the snapshot set defined in the light grey region, hence the size  $N$  of each snapshot, as described in chapter 3, is eight times the number of mesh points in this zone. Once trained the POD model, the surrogate response on the FOM/ROM boundary interface (blue curves in the figure) is extracted and used as boundary conditions to iterate the full order CFD solver in the inner domain (orange domain).



**Figure 5.3:** Zonal approach, FOM/ROM domains

In order to properly impose a specific boundary condition, the ZEN flow solver adopts the ghost-cell method that consists in defining an artificial layer of additional cells across the boundary surface and assigning to the ghost layer variables specific





**Figure 5.4:** Boundary condition setting

values which satisfy the required condition on the boundary faces. This is needed to calculate the right flow fluxes at a boundary cell face. Hence, a peculiar boundary condition type is selected for the ZEN flow solver. With reference to figure 5.4, which represents a zoom of figure 5.3 around a generic computational cell across the domain interface, the solution vector at the interface is derived from the POD surrogate approximation on each common point  $i$  which is shared by the cells of the two domains. Therefore, the solution vector in the fictitious external ghost-cell adjacent to the FOM internal cell  $(i, 1)$ , is computed as follows

$$S_{(i,0)}^{FOM} = S_{(i,1/2)}^{ROM} + S_{(i+1,1/2)}^{ROM} - S_{(i,1)}^{FOM} \quad (5.2)$$

In this way, depending on the actual value  $S_{i,1}^{FOM}$  computed by the CFD code in the first FOM domain cell during the iterations, the solution  $S_{(i,0)}^{FOM}$  in the ghost-cell is adjusted to satisfy the boundary condition imposed on the face  $(i, 1/2)$  and derived from the surrogate model. This guarantees to always find the POD solution on the interface boundary.

The application of the zonal approach to the surrogate model, hence, requires some further steps in addition to those described in the previous section. Given a design site, it is needed to

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

- apply equation 4.23 to predict the CFD flow field in the external ROM domain;
- extract the predicted flow field on the interface between FOM/ROM domains;
- compute the full order, iterative CFD solution in the internal FOM domain by imposing the extracted boundary condition on the boundary of the inner domain.

With such an approach, computing any aerodynamic coefficient or surface distribution of interest (e.g., pressure or skin friction distributions) is a mere post-processing operation, as they can be directly drawn out by the CFD solver output.

### 5.5 Error analysis

In order to compare the performance of the POD surrogate model with respect to the full order CFD model, a validation sampling is defined and a set of statistics are computed. Each surrogate model, in fact, has to be assessed not in the same design sites used to train it. Indeed, the goal is to evaluate the potential of the model to globally approximate the design space. A new DOE sampling of size  $\bar{M}$  is performed on the 16-dimensional design space and the new design sites are evaluated by means of both the surrogate and the full order model. Then, the aerodynamic efficiency  $\frac{C_L}{C_D}$  is computed and used to assess the following error statistics:

- the percentage error  $PE(i), i = 1, \dots, \bar{M}$

$$PE(i) = \left| \frac{\frac{C_L}{C_D} - \frac{\hat{C}_L}{\hat{C}_D}}{\frac{C_L}{C_D}} \right| \times 100 \quad (5.3)$$

- the mean percentage error

$$MPE = \frac{1}{\bar{M}} \sum_{i=1}^{\bar{M}} PE(i) \quad (5.4)$$

- the standard deviation of the percentage error

$$SDPE = \frac{1}{\bar{M} - 1} \sum_{i=1}^{\bar{M}} [PE(i) - MPE]^2 \quad (5.5)$$

- the  $R$ -squared metric

$$R^2 = 1 - \frac{\sum_{i=1}^{\bar{M}} \left( \frac{C_L}{C_D} - \frac{\hat{C}_L}{\hat{C}_D} \right)^2}{\sum_{i=1}^{\bar{M}} \left( \frac{C_L}{C_D} - \frac{1}{\bar{M}} \sum_{i=1}^{\bar{M}} \frac{C_L}{C_D} \right)^2} \quad (5.6)$$

where index  $i$  denotes the  $i$ th sample of the validation DOE plan, the hat quantities refer to the surrogate predictions while the hat-less to the full order ones. These statistics, however, give an outlook of the propagation of the POD model prediction error on the airfoil surface in an integral form, because only aerodynamic force coefficients appear. Even if such a kind of error analysis is very useful to understand the suitability of the surrogate model to approximate the fitness function in an aerodynamic optimization process, which usually requires the evaluation of aero-coefficients, it does not provide a definitive measure of the model accuracy. To this end, the point-to-point relative error between the exact CFD computations of the validation plan and the predicted values of the POD model is introduced at snapshot level:

$$Er(i) = \frac{1}{N} \sum_{j=1}^N \left| \frac{S_i(j) - \hat{S}_i(j)}{S_i(j)} \right| \times 100 \quad (5.7)$$

where  $N$  is the snapshot size,  $S_i(j)$  is the  $j$ -th element of the snapshot built on the exact CFD solution at the  $i$ -th validation design site and  $\hat{S}_i(j)$  is the predicted solution of the  $i$ -th validation design site.

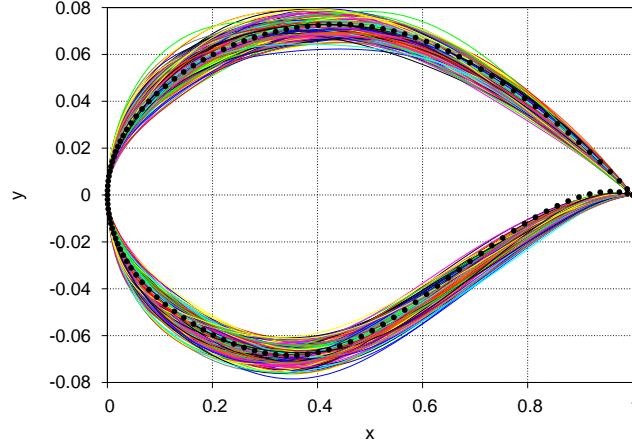
## 5.6 Validation of POD/ROMs in transonic flow

An initial DOE Latin Hypercube sampling is performed on the 16-dimensional design space as defined in section 5.1. The size  $M$  of the initial sampling is 180 to cover each design variable with a sufficient number of samples. The obtained design sites  $w_i$  are then translated into profile geometry thanks to the equation set 3.8, 3.9 and 3.9. 180 volume mesh generations and CFD computations are launched in parallel at fixed flow conditions to compute the flow fields around the airfoil shapes. As mentioned above, the baseline geometry is a modified RAE 2822 airfoil, scaled to 14% thickness ratio to enhance transonic flow effects. Figure 5.5 shows the whole set of obtained airfoil shapes compared to the baseline airfoil (black dots). The corresponding pressure and

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

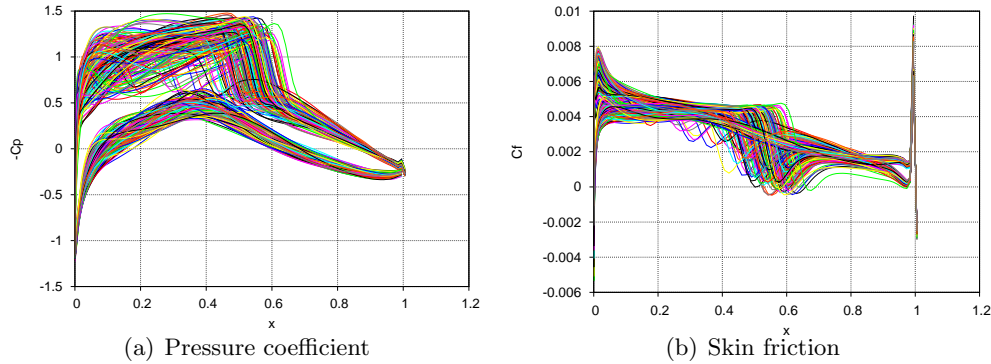
skin friction distributions are reported in figures 5.6(a) and 5.6(b): it is clearly evident that the range of shock wave locations on the suction side and the regions of separated flows are very different and distributed within the dataset.



**Figure 5.5:** 180-sized snapshot airfoils

The flow conditions are summarized in Table 5.2. Fully turbulent flow is assumed. For each airfoil shape, a single-block structured volume mesh made of 25186 points (12288 cells) is computed through an automatic hyperbolic grid generator tool. The first cell at the wall is placed so as to always have a unit  $y^+$  at the specified flow conditions. Figure 5.7 shows a sketch of the standard quality volume mesh around the baseline airfoil.

The FOM/ROM domains are then defined to assess the POD surrogate model. With reference to figure 5.8, the  $d$  parameter is introduced as the distance of the FOM/ROM

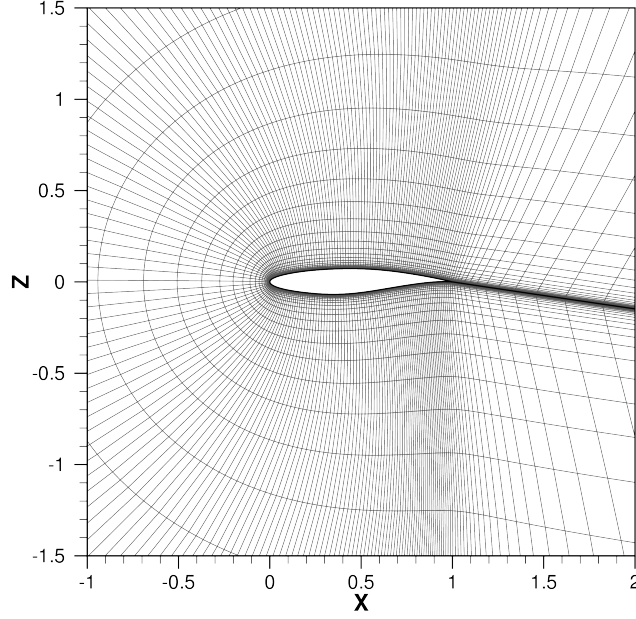


**Figure 5.6:** 180-sized snapshots ensemble

## 5.6 Validation of POD/ROMs in transonic flow

Mach number	Reynolds number	Angle of attack [ $^{\circ}$ ]
0.729	6.5e+06	2.0

**Table 5.2:** Design point



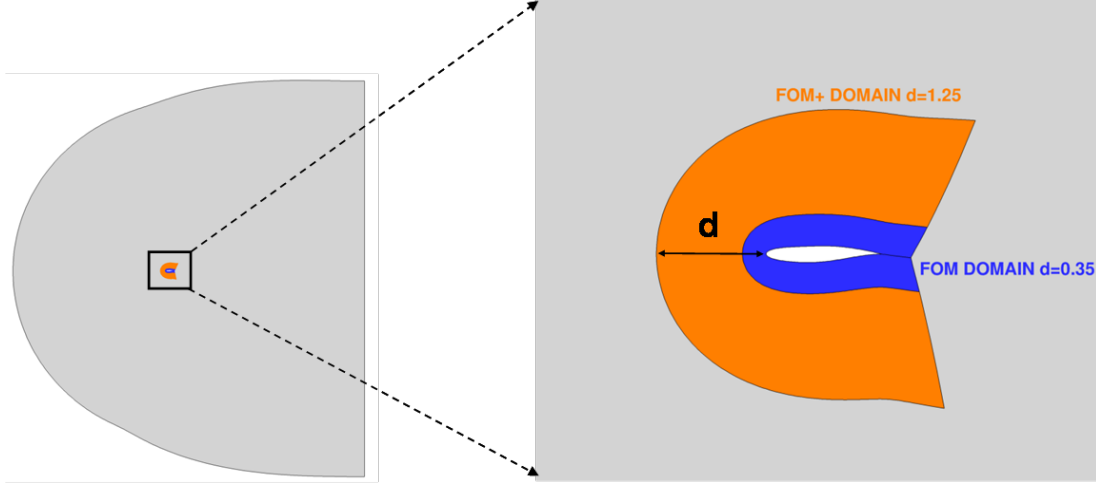
**Figure 5.7:** Volume mesh around the airfoil

interface from the airfoil leading edge; indeed, different POD-based reduced order models can be defined by varying this distance and, hence, reducing or increasing not only the size but also the intrinsic variability of the snapshot set. It is straightforward that the farther the ROM domain will be placed from the airfoil, the smaller the mean variation of the CFD field. At this aim, as eight heterogeneous variables (spatial coordinates, density, pressure, etc..) are condensed in the same snapshot and each of them has its own range of variation, a set of scaling factor for each of the eight variables is defined and applied prior to feed the POD model in order to avoid mistakes in the correlation process. In fact, it could happen that the POD reduction would give more relative importance to the snapshot variables which exhibit the biggest absolute values or the widest range of variation. To avoid this, the scaling factors are designed so as to approximately bring the range of variation of each variable to the interval  $[0,1]$ . Of course, in the present investigation the scaling factors are defined once and kept

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

constant, even when varying the ROM domains and the snapshot sizes.



**Figure 5.8:** FOM/ROM domains with varying interface

Several surrogate models (here referred to as “SM”) are then trained on the computed and scaled CFD set:

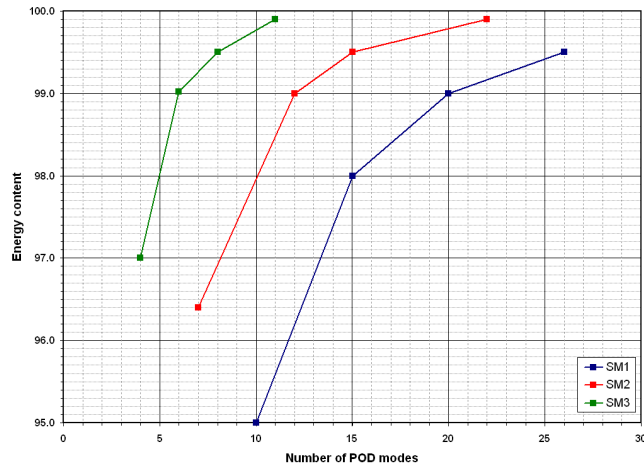
- SM1 is a POD surrogate model with  $d = 0$ , i.e. no FOM domain is defined, the ROM domain coincides with the full domain and no boundary condition is exchanged. The snapshot size  $N$  is 201488;
- SM2 is a POD surrogate model with  $d = 0.35$ , i.e. the FOM domain is the blue one in figure 5.8. The snapshot size  $N$  is 91792
- SM3 is a POD surrogate model with  $d = 1.25$ , i.e. the FOM domain is the orange one in figure 5.8. The snapshot size  $N$  is 75232;
- SM4 is a Kriging interpolation response surface on  $C_L/C_D$ , i.e. a standard meta-model directly trained on a selected objective function;
- SM5 is a quadratic polynomial regression response surface on  $C_L/C_D$ . Given  $p$  the number of design variables, at least  $(p + 1) \times (p + 2)/2$  design sites should be evaluated to train this type of model. In the present case,  $(p + 1) \times (p + 2)/2 = 153$ , hence the size of the *a-priori* sampling is sufficient.

SM1, SM2 and SM3 are POD-based reduced order models, while SM4 and SM5 are introduced on purpose to compare the presented methodology with standard meta-models. The ensemble energy content threshold  $\epsilon$  is reported in figure 5.9(a) as a function of the number of POD modes for each of the POD-based approximations. It is clearly evident that the heaviest model (SM1) requires a big number of modes even to reproduce a relatively low energy level (95%), while SM3 performs considerably better (97%) with just 4 modes preserved. It must be underlined that SM2 requires more modes with respect to SM3 probably because the corresponding domain embeds part of the supersonic region on the airfoils suction side: figure 5.9(b) clarify this issue as it reports the FOM/ROM domains (as in figure 5.8) superimposed with the local Mach number contours. The solution is here computed around an airfoil selected within the ensemble database. While SM3 ROM domain is quite far off the supersonic region, the SM2 FOM/ROM interface lies across it, thus introducing a stronger source of variability (and of slight discontinuity due to the shock wave) into the ensemble. However, in order to make a fair comparison, hereinafter the minimum number of modes which assures at least a 95% threshold of the ensemble energy will be preserved for each model, i.e. 10 modes for SM1 (95%), 7 modes for SM2 (96.4%) and 4 modes for SM3 (97%). In other words, the models will be compared not on a pre-determined number of modes basis, but on a preserved energy basis.

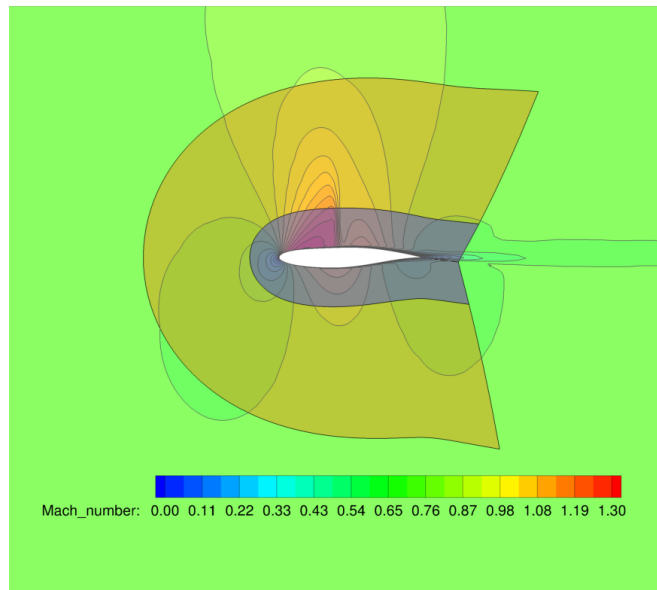
A validation plan is generated with a new LHS sampling of size 50. The goodness-of-fit for each of the 5 models is estimated by means of the statistics described in section 5.5 and involving the aerodynamic efficiency as target function. Once more, it is stressed here that, while models SM1, SM2 and SM3 are POD-based and hence physics-based, SM4 and SM5 are meta-models of the aerodynamic efficiency function, trained on the high-fidelity results computed on the *a-priori* DOE sampling and interpolated on the design sites of the validation plan. Table 5.3 summarizes the results. A clear information can be drawn about an ideal ranking of the tested models: SM3 exhibits superior performances for each estimation parameter, while the quadratic polynomial fitting is completely unfit to approximate the objective function. It must be noted that SM3 performs very well even on the SDPE estimate (eqn. 5.5) which gives an idea of how big is the variation of the percentage prediction error along the validation sampling: the very low value found means that the prediction error at any design site is approximately the same and close to the mean value. This is a very desirable feature

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---



(a) Energy content vs number of POD modes



(b) FOM/ROM interface cutting and embedding the expansion lobe

**Figure 5.9:** Effect of zonal interface on the energy amount captured by POD



## 5.6 Validation of POD/ROMs in transonic flow

for a surrogate model designed for an optimization framework. On the other hand, SM5 shows very poor performances because such a polynomial regression is unable to approximate a multi-modal, rapidly changing objective function.

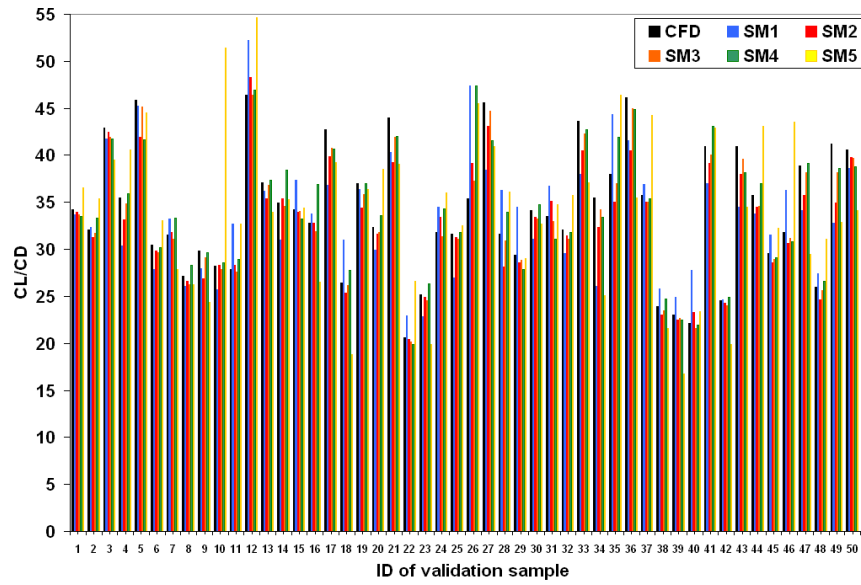
Surrogate	R-squared	MPE	SDPE	Ranking
SM1	0.5876	10.33	48.14	4
SM2	0.8899	4.55	12.85	2
SM3	0.9791	2.30	1.61	1
SM4	0.8657	4.56	26.61	3
SM5	0.06074	15.62	171.62	5

**Table 5.3:** Surrogate goodness-of-fit estimation

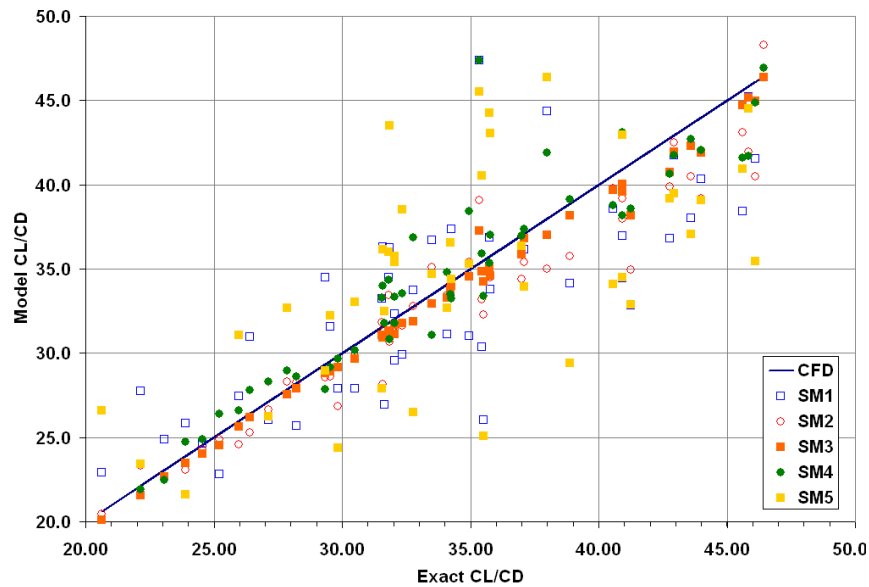
An interesting point comes out by comparing SM2 and SM4, which exhibit very similar error indices: as Kriging is a standard meta-model used in aerodynamic data reconstruction, it can be stated that the POD surrogate model accuracy increases by moving the FOM/ROM interface away from the airfoil surface and there exists a peculiar value of the distance  $d$  for which its predictive power is very close to standard and efficient interpolation techniques. To give a graphic idea of the figures reported in table 5.3, figure 5.10(a) shows a bar plot representing the full order CFD and the surrogate models predictions of aerodynamic efficiency as a function of the validation set index (ranging from 1 to 50). The plot confirms that SM2 and SM3 follow almost exactly the trends of the exact CFD data, so preserving the monotonicity of the sampling set. Moreover, SM1, SM2 and SM3 are able to capture the airfoil shapes with both the best (ID 12) and the worst (ID 22) aerodynamic efficiency in the validation dataset. Figure 5.10(b) reports the correlation plot between the models prediction and “true” CFD data. Again, SM2, SM3 and SM4 are globally closer to the linear trend, resulting in a better fit. The correlation plot highlights another significant feature of SM3 model, as it generally underestimates the aerodynamic efficiency. For further comparisons, table 5.4 summarizes the validation set indices where each model predicts the highest and lowest efficiency, the corresponding values of aerodynamic efficiency and the percentage error with respect to the CFD datum. This is useful to evaluate the capability of the model to identify the global extrema of the objective function. It is observed that only SM4 leads to a wrong estimation of the position of the “optimal” airfoil while SM5 tends to overestimate the worst profile.

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---



(a) Bar plot



(b) Correlation plot

**Figure 5.10:** Surrogate models predictions of  $CL/CD$  on the validation plan

## 5.6 Validation of POD/ROMs in transonic flow

Surrogate	ID of max	ID of min	max	min	$\Delta\text{max}$ (%)	$\Delta\text{min}$ (%)
CFD	12	22	46.43	20.61	0	0
SM1	12	22	52.19	22.84	12.40	10.81
SM2	12	22	48.27	20.45	3.95	-0.77
SM3	12	22	46.40	20.12	-0.07	-2.39
SM4	26	22	47.40	19.86	2.08	-3.65
SM5	12	39	54.62	16.78	17.63	-18.59

**Table 5.4:** Surrogate estimations of aerodynamic efficiency for best and worst validation airfoils

The last two properties, i.e. the capability to preserve the monotonicity of the dataset and to correctly identify the best/worst candidates, are crucial aspects in surrogate-based optimization (SBO), so that models SM2 and SM3 seem to be more suitable for this purposes. A qualitative comparison is also proposed at airfoil surface distribution level. Figure 5.11(a) and 5.11(b) show the pressure coefficient distribution as obtained from CFD computation and surrogate models SM1, SM2 and SM3 respectively on the most (airfoil ID 12) and the least (airfoil ID 22) efficient airfoil. Skin friction distributions are also compared in figure 5.12(a) and 5.12(b). Airfoil ID 12 is featured with a double shock structure on the upper surface which is not really captured by SM1, partially captured by SM2 and fully captured by SM3. On the other hand, a strong shock wave and a shock-induced separation characterize quite a complex aerodynamics for airfoil ID 22, but all the models rather show a pretty good prediction for this case. This outwardly strange behaviour is probably motivated by the fact that the training database contains more than one CFD solutions which present aerodynamic features similar to ID 22, so that even less accurate models provide a satisfying prediction. On the contrary, airfoil ID 12 represents a quite unique sample. Two more solutions (ID 20 and 35) have been selected from the validation plan and shown for comparison in figure 5.11(c) and 5.11(d) (pressure distributions) as well as figures 5.12(c) and 5.12(d) (skin friction). The smearing out and dislocation of pressure jumps across the shock wave can be identified as clear features and peculiar characteristics of SM1. POD models accuracy is finally evaluated and compared in terms of the point-to-point snapshot percentage prediction error given by equation 5.7. Figure 5.13 shows the results for each snapshot belonging to the validation plan (again ranging from 1 to

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

50). The error index is plotted in logarithmic scale because SM1 exhibits 1-2 order of magnitude larger errors than the others. As the snapshot prediction error takes into account the reconstruction error as a whole, this is probably the most complete and global comparison possible: as a result, it comes out that, in strong transonic conditions, training a POD model on the full CFD domain (SM1) would lead to misleading results in the prediction phase, as the model would not be able to catch the highly non linear trends which characterize this kind of flows. Indeed, the high number of POD modes required and the low goodness-of-fit performance suggest that further modelling is needed to adapt the computation of the basis vectors and modal coefficients to transonic aerodynamics. In the next sections, we will introduce some adaptive sampling concepts to globally improve the reduced order models predictions.

The final comparison is made in terms of POD model accuracy versus computational time and cells saving. In particular, the R-squared prediction error is taken as a measure of the model accuracy, while the time saving index ( $TS$ ) and the cells saved index ( $CS$ ) are defined as

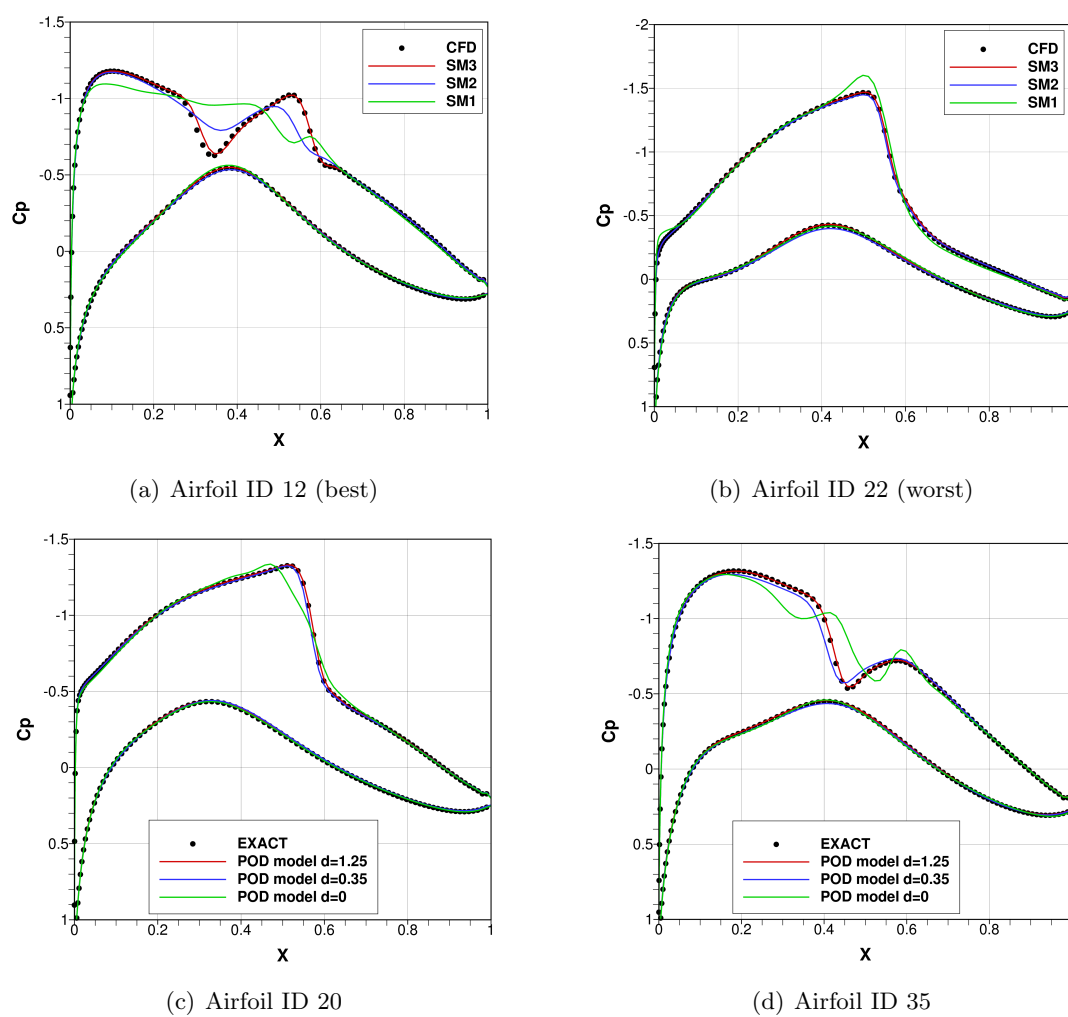
$$TS = \frac{T_{FULL} - T_{SM}}{T_{FULL}} \quad (5.8)$$

$$CS = \frac{N_{SM}}{N_{FULL}} \quad (5.9)$$

where  $T$  and  $N$  are respectively the computational time for 1000 CFD iterations and the number of solved computational cells. The subscripts  $FULL$  and  $SM$  refer to the full grid CFD computation and the CFD computation on the smaller FOM domain. In figure 5.14 the three indices are plotted against the distance  $d$  of the FOM/ROM interface from the airfoil leading edge. It shows that a clear trade-off exists between accuracy and time/cells saving and provides useful guidelines to tailor the choice of the best POD model to the basic requirements of the target application. For instance, if the target is to do a pre-screening of the objective space, one could use a faster and less accurate POD model which however guarantees the preservation of the physics.

### 5.7 Adaptive sampling

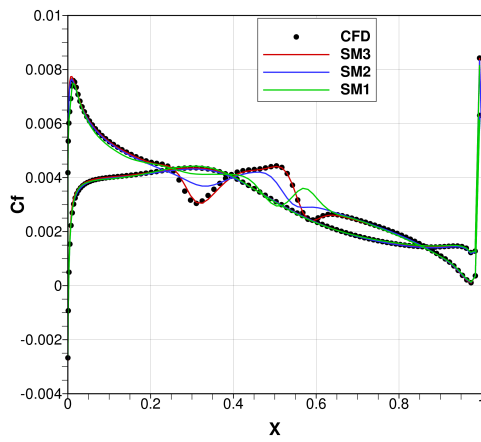
In previous sections, it was shown that, provided an initial ensemble of flow solutions, feeding the POD model with the full flow fields leads to an inaccurate prediction on the validation sampling, i.e. on design sites which do not belong to the initial ensemble. As



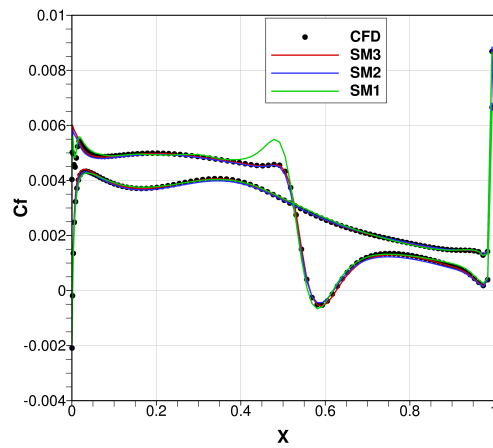
**Figure 5.11:** Pressure coefficient comparison

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

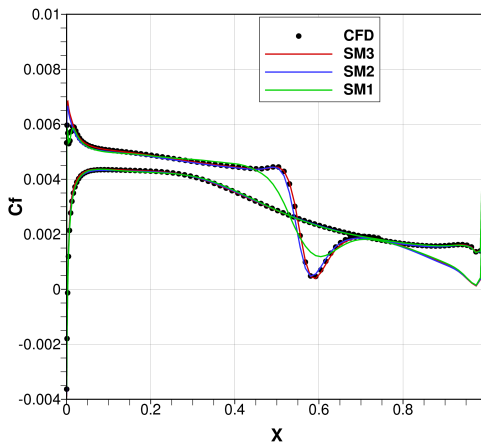
---



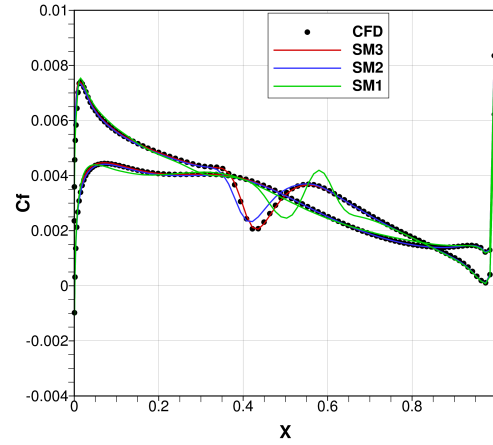
(a) Airfoil ID 12 (best)



(b) Airfoil ID 22 (worst)

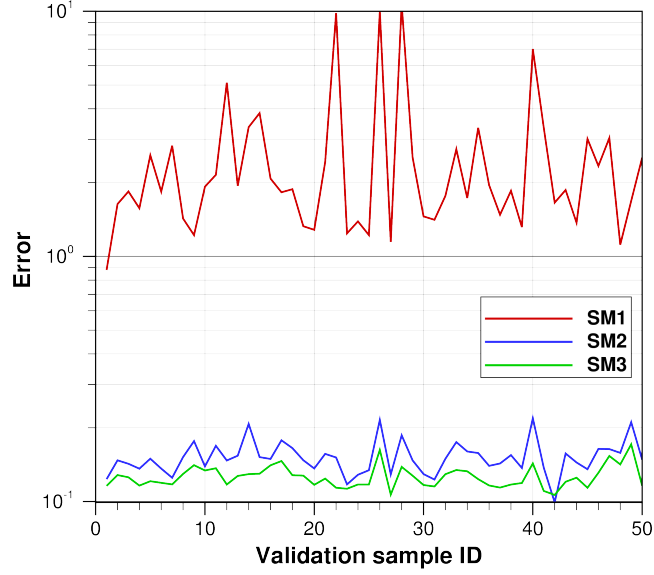


(c) Airfoil ID 20



(d) Airfoil ID 35

**Figure 5.12:** Skin friction coefficient comparison



**Figure 5.13:** Snapshot error prediction

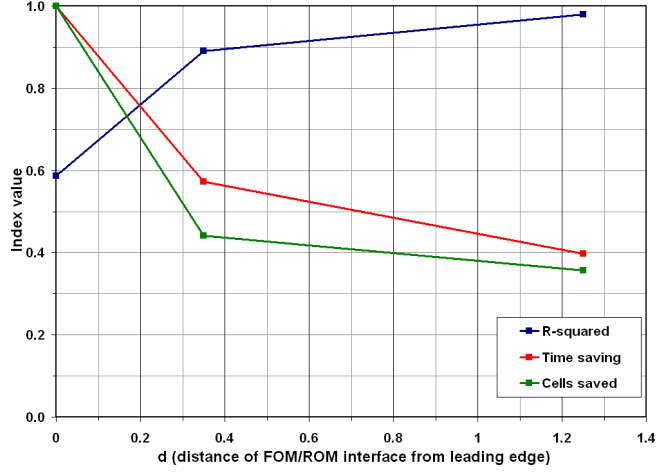
the need to compute the fitness of new design candidates in the design space is usually encountered during an explorative search process, some improvements are needed to recover this issue. In the proposed example, we selected a 180-sized *a-priori* ensemble to sample the 16-dimensional design space, but which is in principle the correct choice of the sample to properly feed the reduced order model? Actually, we do not have an answer. Indeed, we do not know neither the right size nor the right locations of the sample points. Intuitively, we would like to have a sampling strategy which would fill the space in an efficient manner and would allocate more points in the regions of the design space where the simulation response is strongly non-linear. Further questions are

- For a given computational budget, can I improve the quality of the POD surrogate (full or zonal) by “intelligently” choosing the training samples?
- For a given POD model quality, can I get the same accuracy level with less high-fidelity computations?

Adaptive sampling strategies can be properly designed to account for these requirements by fulfilling so-called “in-fill” criteria. While *a-priori* sampling techniques do

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---



**Figure 5.14:** Trade-off accuracy vs resource saving

not use any information about the model prediction, adaptive techniques incrementally select new sampling points by exploiting the input/output relation observed at the previous stage. However, they should be tailored to the particular model which has to be improved in order to exploit its own predictions and characteristics. Hence, we will propose some adaptive DoE strategies for POD-based reduced order models. We will refer to the joint work with Goblet et. al (61), developed within the Clean Sky JTI-GRA *ROM&O* Project, funded by the EU Seventh Framework Programme, Proposal Number 255779, and answering the call for proposal JTI-CS-2009-1-GRA-05-004. Generally speaking and with reference to the nomenclature used in chapter 4, the quality of the POD/ROM models basically depends on: (1) the quality of the modal basis  $\{\phi_1, \dots, \phi_{\hat{M}}\}$ ; (2) the quality of the modal coefficient RBF models  $\{\alpha_1, \dots, \alpha_{\hat{M}}\}$ . Indeed, they both depend on the choice of the snapshot dataset. In the following, two methods are proposed to properly balance the improvement of the POD model and the space-filling properties. Both methods are based on the leave-one-out cross-validation technique.

### 5.7.1 Improvement of the modal basis

The first method is aimed at improving the modal basis, which represents the core of the POD modelling. Indeed, given a POD model built on a snapshot ensemble  $\{S_1(w_1), \dots, S_M(w_M)\}$ , we want to find a new point  $w_{new}$  in the design space to be



included in the previous sampling set so that the new POD model basis, built on the new set  $\{S_1(w_1), \dots, S_M(w_M), S(w_{new})\}$ , will result in improved predictions and better exploration of the design space at the same time. First, the relative influence of the  $j^{th}$  snapshot on the modal basis is computed as

$$\text{Infl}_{\text{Basis}}^{\text{Rel}}(w_j) = \frac{\text{Infl}_{\text{Basis}}(w_j)}{\sum_{k=1}^n \text{Infl}_{\text{Basis}}(w_k)} \quad (5.10)$$

where

$$\text{Infl}_{\text{Basis}}(w_j) = \sum_{i=1}^{\hat{M}} \sigma_i \left( \frac{1}{|(\phi_i^T, \phi_i^{-j})|} - 1 \right) \quad (5.11)$$

is the influence of the  $j^{th}$  snapshot on the modal basis and  $\phi_i^{-j}$  is the  $i^{th}$  column vector of  $U^{-j}$  and  $U^{-j}\Sigma^{-j}V^{-jT}$  is a thin SVD of the matrix

$$(S_1 - \bar{S} \quad \dots \quad S_{j-1} - \bar{S} \quad \mathbf{0} \quad S_{j+1} - \bar{S} \quad \dots \quad S_n - \bar{S})$$

The scalar product  $(\phi_i^T, \phi_i^{-j})$ , as usual, gives the projection of one of the two vectors on the other, hence if they are almost orthogonal, the quantity will be zero and the influence term will go to infinity, while if they are almost parallel, the influence contribution will be almost zero. The relative influence is normalized with the sum of the influence of the whole set of snapshots and weighted with the singular values as they reflect the importance of each mode with respect to the whole basis. A priori the computation of these quantities would require the computation of  $M$  thin SVD of  $N \times M$  matrices where  $N = v \times q$  (number of flow variables  $\times$  number of mesh points) can be huge as it is related to the dimension of the data set. As detailed in (61), it is possible to get a cheaper evaluation by computing  $M$  thin SVD on  $M \times M$  matrices. Once known the relative influence of each snapshot on each modal basis vector, we need to decide where and how to choose the new sampling point. From a theoretical point of view, we would need to sample “near” the design site where the relative influence is the highest. But two questions arise: what does it mean “near” exactly? And, if sampling near a known design site, what about exploring new undiscovered regions? To answer the first questions, the design space is heavily sampled with LHS technique, e.g. 100 times the dimension  $p$  of the design space. Then, the Euclidean distance of each new sampled point  $y_i, i = 1, \dots, t = 100p$  from each of the snapshot sites  $w_k, k = 1, \dots, M$  is computed and, for each  $y_i$ , the distance from the nearest snapshot  $w_{\bar{k}}$  is stored as  $d(w_{\bar{k}}, y_i)$ .

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

The second question is more subtle as it concerns the trade-off between locally accuracy and design space global exploration. Knowing the relative distances between new sampling points and snapshot sites, we would also like to sample far away from the known points in order to potentially enrich the global prediction of the POD model. The final choice is to weight the distance  $d(w_{\bar{k}}, y_i)$  with the relative influence of the  $\bar{k}^{th}$  snapshot on the modal basis and compute a potential of enrichment  $V_\phi$  for each new candidate  $y_i$

$$V_\phi(y_i) = d(w_{\bar{k}}, y_i) \text{Inf}_{\text{Basis}}^{\text{Rel}}(w_{\bar{k}}) \quad (5.12)$$

Finally, the first adaptive strategy is to choose  $w_{\text{new}} = \text{argmax}_{y_i} V_\phi(y_i)$ .

### 5.7.2 Improvement of the coefficients models

As already described in chapter 4, the POD modal coefficients are provided with a global approximation character through radial basis functions. The second adaptive method is then conceived to improve the quality of these RBF models. Given a POD model built on a snapshot ensemble  $\{S_1(w_1), \dots, S_M(w_M)\}$ , we want to find a new point  $w_{\text{new}}$  in the design space to be included in the previous sampling set so that the new modal coefficients models, built on the new set  $\{S_1(w_1), \dots, S_M(w_M), S(w_{\text{new}})\}$ , will result in improved predictions and explore the design space at the same time. Two secondary strategies are proposed: the first aims at improve the prediction when the quality of the coefficient models are not comparable, the second is designed to work well when the coefficient models show similar levels of accuracy.

#### 5.7.2.1 First strategy

This strategy is applied when one of the coefficient model  $\alpha_1(w), \dots, \alpha_M(w)$  exhibits low quality with respect to the others. For the sake of clarity, according to the nomenclature used in chapter 4, we will denote as  $\alpha_i(w)$  the  $i^{th}$  coefficient model and with  $\alpha_i^{(j)}$  the  $i^{th}$  coefficient corresponding to the  $j^{th}$  snapshot in the ensemble database. First of all, we need to evaluate the quality of the coefficient model, possibly taking into account the relative importance of the mode itself. This is done by computing the correlation coefficient of the model  $\alpha_i(w)$  as

$$\text{Corr}(\alpha_i) = \frac{M \sum_{j=1}^M \alpha_i^{(j)} \alpha_i^{-j}(w_j) - \sum_{j=1}^M \alpha_i^{(j)} \sum_{j=1}^M \alpha_i^{-j}(w_j)}{\sqrt{M \sum_{j=1}^M (\alpha_i^{(j)})^2 - \left(\sum_{j=1}^M \alpha_i^{(j)}\right)^2} \sqrt{M \sum_{j=1}^M \alpha_i^{-j}(w_j)^2 - \left(\sum_{j=1}^M \alpha_i^{-j}(w_j)\right)^2}} \quad (5.13)$$

where  $\alpha_i^{-j}(w)$  is the model built on the left- $j^{th}$ -out correspondence

$$\{w_1, \dots, w_{j-1}, w_{j+1}, \dots, w_M\} \rightarrow \{\alpha_i^{(1)}, \dots, \alpha_i^{(j-1)}, \alpha_i^{(j+1)}, \dots, \alpha_i^{(M)}\} \quad (5.14)$$

i.e. subtracting one by one the  $j^{th}$  design site and coefficient from the ensemble and recomputing the RBF model. The correlation coefficient (5.13) provides a statistical index which measure the accuracy of each model. To further take into account the energy contribution of the  $i^{th}$  mode, we define a “weighted” quality of the model  $\alpha_i(w)$  as  $WQ(\alpha_i) = (\sigma_i / \sum_{j=1}^{\hat{M}} \sigma_j) \text{Corr}(\alpha_i)$ . As already mentioned, the coefficient model with the lowest weighted quality is selected and denoted with index  $\alpha_{ii}(w)$ . The relative influence of the  $j^{th}$  snapshot on the  $ii^{th}$  modal coefficient is defined as  $\text{Infl}_{\alpha_{ii}}(w_j) = |\alpha_{ii}^{(j)} - \alpha_{ii}^{-j}(w_j)|$ , by leaving out the  $j^{th}$  candidate and using it to estimate the model error. From now on, the procedure is the same as for the modal basis improvement: the design space is heavily sampled with LHS technique, e.g. 100 times the dimension  $p$  of the design space. Then, the Euclidean distance of each new sampled point  $y_i, i = 1, \dots, t = 100p$  from each of the snapshot sites  $w_k, k = 1, \dots, M$  is computed and, for each  $y_i$ , the distance from the nearest snapshot  $w_{\bar{k}}$  is stored as  $d(w_{\bar{k}}, y_i)$ . By weighting the distance  $d(w_{\bar{k}}, y_i)$  with the relative influence of the  $\bar{k}^{th}$  snapshot on the  $ii^{th}$  modal coefficient, a potential of enrichment  $V_{\alpha_{ii}}$  for each new candidate  $y_i$  with respect to the worst model can be computed as

$$V_{\alpha_{ii}}(y_i) = d(w_{\bar{k}}, y_i) \text{Infl}_{\alpha_{ii}}(w_{\bar{k}}) \quad (5.15)$$

Finally the new sampling point is selected at  $w_{\text{new}} = \text{argmax}_{y_i} V_{\alpha_{ii}}(y_i)$ .

The leave-one-out procedure allows to determine the quantities  $\alpha_i^{-j}(w)$ , but its cost is generally high as it would apparently require the building of  $\hat{M} \times M$  new approximations. However, when using RBF network interpolators for POD coefficient models, the leave-one-out procedure can be performed at low-cost by using the efficient formula provided by Rippa (62). Indeed, the author showed that, in order to compute  $|\alpha_{ii}^{(j)} - \alpha_{ii}^{-j}(w_j)|$ , all the needed information are available without any extra cost as the required model parameters have been already computed during the construction of  $\alpha_{ii}(w)$ .

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

### 5.7.2.2 Second strategy

This strategy is used when the quality of the coefficient models are comparable and it is very similar to the improvement of the modal basis. The relative influence of the  $j^{th}$  snapshot on the coefficient models is computed as

$$\text{Infl}_{\text{Coeff}}^{\text{Rel}}(w_j) = \frac{\text{Infl}_{\text{Coeff}}(w_j)}{\sum_{k=1}^n \text{Infl}_{\text{Coeff}}(w_k)} \quad (5.16)$$

where

$$\text{Infl}_{\text{Coeff}}(w_j) = \sum_{i=1}^{\hat{M}} \sigma_i \text{Infl}_{\alpha_i}(w_j) = \sum_{i=1}^{\hat{M}} \sigma_i |\alpha_i^{(j)} - \alpha_i^{-j}(w_j)| \quad (5.17)$$

is the influence of the  $j^{th}$  snapshot on the coefficient models weighted with the corresponding singular values. It must be noted that now we do not need to compute the worst  $\alpha_i$  model as we are supposing to improve them all together. Therefore, the design space is heavily sampled with LHS technique, e.g. 100 times the dimension  $p$  of the design space. Then, the Euclidean distance of each new sampled point  $y_i, i = 1, \dots, t = 100p$  from each of the snapshot sites  $w_k, k = 1, \dots, M$  is computed and, for each  $y_i$ , the distance from the nearest snapshot  $w_{\bar{k}}$  is stored as  $d(w_{\bar{k}}, y_i)$ . By weighting the distance  $d(w_{\bar{k}}, y_i)$  with the relative influence  $\text{Infl}_{\text{Coeff}}^{\text{Rel}}(w_j)$ , a potential of enrichment  $V_\alpha$  for each new candidate  $y_i$  with respect to the POD coefficient models can be computed as

$$V_\alpha(y_i) = d(w_{\bar{k}}, y_i) \text{Infl}_{\text{Coeff}}^{\text{Rel}}(w_{\bar{k}}) \quad (5.18)$$

Finally the new sampling point is selected at  $w_{\text{new}} = \text{argmax}_{y_i} V_\alpha(y_i)$ .

## 5.8 Concluding remarks

Three POD/ROM models have been trained and compared: the first one consisted in feeding the POD ensemble with the full field, hence without any domain decomposition; in the second and third one, the zonal approach was applied by defining two different values of the distance of the interface from the airfoil leading edge. Results showed that the model accuracy is strongly dependent on the distance parameter, mainly because of the presence of the supersonic expansion lobe and the pressure jump across the shock wave on the airfoil suction side. In fact, the model named SM3 showed superior

performances with respect to both the other POD models and standard interpolation techniques like Kriging and regression methods like quadratic polynomial fitting. It also allows to get very accurate reconstruction of airfoil surface distributions and, hence, of aerodynamic coefficients, which are very often the actual target of aerodynamic design. Another important conclusion of the work is that it seems completely misleading to base the POD ensemble on the full flow field when transonic conditions and shape modifications act together. Indeed, as the POD reconstruction is a linear combination of POD modes, capturing the combined non-linear effects of boundary layer and compressibility is hardly possible when the position and intensity of the shock wave and its interaction with the boundary layer vary too much. Globally, the proposed POD surrogate model showed to have many characteristics which make it suitable to aerodynamic design. However, a trade-off was found between POD model accuracy and resource saving as a function of the distance parameter: the smaller the full order domain, the shorter the computational time required but also the less accurate the reconstruction. In order to get rid of these issues, a set of strategies have been proposed to update and enhance the surrogate/POD model through adaptive DOE techniques. Indeed, the selection of the design sites to be included in the POD ensemble, instead of being fully derived from an *a-priori* sampling strategy, can be tailored to match specific POD-related improvement requirements. In the next chapter, the benefits of the adaptive DoE will be assessed in an real-time updating process. Moreover, the inclusion of the POD surrogate model within an optimization framework will be presented to come up with a complete, self-updating surrogate/reduced order optimization method and the resulting coupling strategy will be studied to fully explore the potential of the presented methodology towards complex aerodynamic design cases.

## 5. POD-BASED REDUCED ORDER MODELLING FOR TRANSONIC SHAPE OPTIMIZATION

---

## 6

# POD-based Evolutionary Optimization in Transonic flow

In this chapter, the POD/ROMs and the adaptive sampling techniques described in the previous chapters are combined within an evolutionary optimization loop and used to validate the developed methodology in transonic flow. Several approaches will be proposed, differing between each other in the key ingredients of the methodology: the construction of the POD model (full/zonal approach), the strategy chosen to compute the training sample (*a-priori*, auto-adaptive) and the strategy to exploit the optimization results (one-shot optimization, real-time updating). The goal of the different surrogate-based optimizations is to improve the performances of the scaled RAE2822 airfoil.

## 6.1 The surrogate-based shape optimization framework

The workflow of the Surrogate-based Shape Optimization (SBSO) is depicted in figure 6.1. Basically, it consists of an *a priori* design of experiment module (a Latin Hypercube sampler), the CST parameterization module, an automatic in-house developed hyperbolic mesh generator, the ZEN CFD flow solver, the POD/ROM module, which encloses also the adaptive sampling techniques, and the ADGLIB optimization library. All these modules have been detailed in previous chapters. The integrating platform is an in-house developed software called GAPOD, which controls the various modules through internal or external calls. For example, the geometric modeller, the mesh gen-

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---

erator and the flow solver can be easily replaced as they appear in the text input files and they are invoked through system calls. On the other hand, the POD module and the optimizer are software library modules directly linked with the launcher. Moreover, the POD module can be activated with two modes: a “standalone” mode and a “coupled/zonal” mode (63). The first one is switched on when the surrogate prediction directly provides the objective/constraint functions, e.g. without any further call of the high-fidelity code. This is, for example, the case when the POD model is trained on the whole flow field. However, as the surrogate model only provides a prediction of the mesh and field variables, a properly designed “condensation” procedure is applied to retrieve the integral coefficients like  $C_l, C_d, C_m$ . In particular, given the general definition of the aerodynamic coefficients as in equation 6.1

$$C_{l,d,m} = \frac{1}{S} \int_S F_{l,d,m} dS \simeq \frac{1}{S} \sum_{i=0}^{N_S} F_{l,d,m}^i \Delta S_i \quad (6.1)$$

where  $S$  is the integration surface, a slight modification of the snapshot definition is made as the discrete integrand function ( $F_{l,d,m}^i \Delta S_i$ ) is added to it. The function  $F_{l,d,m}$  represent the pressure and shear stresses distribution on the wall times the scalar product of the unit vector normal to the wall and the unit vector of the axis along which we are projecting the aerodynamic force.  $F_{l,d,m}^i$  and  $\Delta S_i$  are the discretized components of  $F_{l,d,m}$  and  $S$ . Hence, once a design site has been evaluated through the surrogate, a simple sum is needed on the corresponding snapshot elements to get the integral coefficients. The “coupled/zonal” mode is designed to apply the zonal CFD/POD approach. Once computed 1) a POD model and 2) an airfoil geometry from a selected design site, the POD model is evaluated to provide a predicted snapshot. Then, the full volume mesh is calculated and the FOM (inner) flow domain is computed and written in the ZEN code file format. A specifically designed interface detects the FOM/ROM interface in the snapshot definition and writes a boundary condition file suitable for the ZEN solver. Finally, the flow solver is launched in the FOM inner domain with the surrogate-derived boundary condition.

### 6.2 Problem definition

The geometry parameterization has already been described in the previous chapter. The design point is summarized in table 5.2. Here, we define the airfoil shape optimization



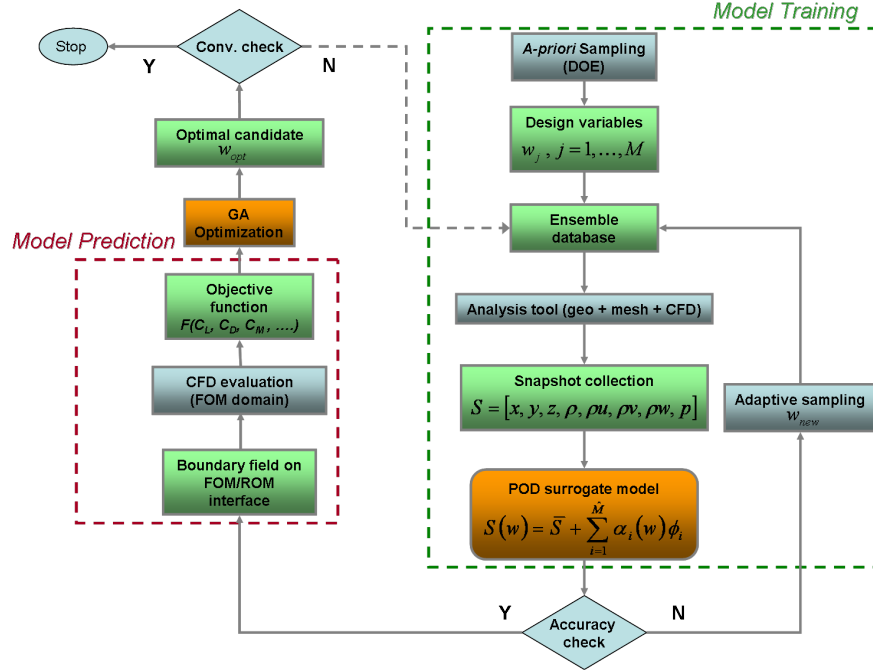


Figure 6.1: Workflow of CFD/POD-based genetic optimization

problem in terms of objective/constraint functions specification

$$\begin{aligned}
 & \underset{w \in DW \subset \mathbb{R}^{16}}{\text{minimize}} && -\frac{C_l}{C_d} \\
 & \text{subject to} && \left(\frac{t}{c}\right)_{\max} = 0.14 \\
 & && C_l \geq 0.5 \\
 & && C_m \geq -0.05 \\
 & && C_m \leq 0.05
 \end{aligned}$$

In other words, the goal is to maximize the aerodynamic efficiency  $\frac{C_l}{C_d}$  while keeping a minimum level of lift generation ( $C_l \geq 0.5$ ) and of pitching moment controllability ( $|C_m| \leq 0.05$ ). Moreover, a geometric constraint is added in order to set the air-foil maximum thickness at 14%: this constraint is automatically satisfied through the parameterization approach, hence it will not appear explicitly. The constraint functions are actually treated as quadratic penalties, hence the constrained optimization is transformed into the following unconstrained problem:

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---

$$\begin{aligned} \underset{W \in Z \subset \mathbb{R}^{16}}{\text{minimize}} \quad & -\frac{C_l}{C_d} + K[\min(C_l - 0.5, 0)]^2 + K[\min(C_m + 0.05, 0)]^2 + \\ & + K[\min(-C_m + 0.05, 0)]^2 \end{aligned} \quad (6.2)$$

where  $K$  is a constant weight (equal to  $10^4$ ) which amplifies the relative importance of possible constraint violations. For instance, a unit penalty will be applied to the objective function in the case of an airfoil having a pitching moment of  $\pm 0.06$ .

### 6.3 Optimization strategies and set up

Several optimization approaches have been set up and tested in order to possibly cover all the issues concerning surrogate/ROM training and prediction. Table 6.1 summarizes the characteristics of each optimization in terms of: fitness evaluator, optimization algorithm, POD energy threshold (when using POD as surrogate), the high-fidelity computational budget, i.e. the total number of calls to the ZEN RANS solver during the optimization, the number  $M_{apr}$  of *a-priori* LHS samples, the number  $M_{adp}$  of adaptively added samples through the strategy defined in the previous chapter and the number  $M_{opt}$  of surrogate-based optima which are iteratively added to the ensemble database. It must be noted that not all the optimization strategies use POD as surrogate: in particular, optimization KGA and EGO have been performed by respectively using a Kriging method as fitness evaluator and the EGO (Efficient Global Optimization) algorithm, based on Kriging and Expected Improvement evaluation, to compute new optimal samples. The EGO algorithm has been described in detail in the literature review as one of the modern standard methods in global optimization.

In the following with the term “truth” or “true” we will indicate the results coming from the high-fidelity CFD solver. Each optimization method is here described in details:

- **DGA**: a plain, brute-force genetic optimization with the full high-fidelity solver ZEN called as fitness evaluator;
- **FPGA1**: a surrogate-based optimization where the aerodynamic analysis is carried out through a POD model built on the complete flow field of a set of 180 initial samples. This case corresponds to the POD-driven “standalone” mode and the surrogate POD evaluator is the one presented as SM1 in chapter 5. No

### 6.3 Optimization strategies and set up

Opt Tag	Fitness evaluator	Optimizer	POD energy	Budget hi-fi	$M_{apr}$	$M_{adp}$	$M_{opt}$
DGA	ZEN	ADGLIB	-	9600	0	0	0
FPGA1	standalone POD	ADGLIB	85%	180	180	0	0
FPGA2	standalone POD	ADGLIB	95%	180	180	0	0
FPGA3	standalone POD	ADGLIB	99%	180	180	0	0
MPGA1	zonal POD	ADGLIB	95%	180	180	0	0
MPGA2	zonal POD	ADGLIB	99%	180	180	0	0
KGA	Kriging	Dakota SOGA	-	180	180	0	0
EGO	Kriging	Dakota EGO	-	553	153	400	-
AFPGA1	standalone POD	ADGLIB	99%	96	32	16	48
AFPGA2	standalone POD	ADGLIB	99%	96	16	32	48
AFPGA3	standalone POD	ADGLIB	99%	96	4	44	48
AMPGA1	zonal POD	ADGLIB	99%	112	8	56	48
AMPGA2	zonal POD	ADGLIB	99%	96	8	40	48

**Table 6.1:** Optimization approaches

zonal approach is used. The POD energy content is 85%. The snapshot size  $N$  is 201488;

- **FPGA2,FPGA3:** same as **FPGA1**, but the POD models are defined by sweeping the energy content (95% and 99%, respectively);
- **MPGA1:** a surrogate-based optimization where the zonal CFD/POD model described in chapter 5 is trained on the same initial design space sampling (180 snapshots) and adopted as objective function evaluator throughout the optimization cycle. The FOM domain is defined as the orange one in figure 5.8 at a distance  $d = 1.25$  chord length from the airfoil leading edge. The POD model used here has been already validated as SM3 in chapter 5. The POD energy threshold is set at 95%. The snapshot size is 75232;
- **MPGA2:** same as **MPGA1**, but the POD energy content is increased up to 99%;
- **KGA:** a surrogate-based optimization where a kriging meta-model, built on the objective function, is coupled to the genetic optimization. Here, the DAKOTA package (64) is used both for optimization process control and algorithm capabilities. The JEGA library (65) was used for optimization purposes. In particular,

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---

the Single-Objective Genetic Algorithm (SOGA) was used to perform optimization on a single objective function with general constraints. The basic notion that underpins kriging is that the sample response values exhibit spatial correlation, with response values modelled via a Gaussian process around each sample location (i.e., samples taken close together are likely to have highly correlated response values, whereas samples taken far apart are unlikely to have highly correlated response values). A Kriging technique has been chosen here because they have been widely used due to their ability to accommodate irregularly spaced data, their ability to model general surfaces that have many peaks and valleys, and their exact interpolation of the given sample response values. The Kriging is initially trained on the usual 180-sized dataset. Then, a surrogate-based iterative optimization scheme is performed consisting in the following steps:

1. adding points to the sample set used to create the surrogate;
2. rebuilding the surrogate;
3. performing a global optimization on the new surrogate;
4. finding of minimizers of the surrogate model;
5. passing a selected optimal subset (in the present case, just the optimum candidate) to the next iteration;
6. re-evaluation of the surrogate points with the “truth” (CFD) model;
7. adding to the set of points upon which the next surrogate is constructed and return back to 1.

This procedure offers a more accurate surrogate to the minimizer at each subsequent iteration, presumably driving to optimality quickly. In the present optimization, 10 SBO iterations are performed.

- **EGO**: a stochastic response surface approximation for the objective function is developed based on some sample points from the “true” CFD simulation. The particular response surface used is a Gaussian process (GP). The GP allows one to calculate the prediction at a new input location as well as the uncertainty associated with that prediction. The key idea in EGO is to maximize the Expected Improvement Function (EIF). The EIF is used to select the location at which a

new training point should be added to the Gaussian process model by maximizing the amount of improvement in the objective function that can be expected by adding that point. A point could be expected to produce an improvement in the objective function if its predicted value is better than the current best solution, or if the uncertainty in its prediction is such that the probability of it producing a better solution is high. Because the uncertainty is higher in regions of the design space with few observations, this provides a balance between exploiting areas of the design space that predict good solutions, and exploring areas where more information is needed. The general procedure used here is:

- build an initial Gaussian process model of the objective function on a initial dataset. For the 16 variable case, 153 design sites are automatically generated by the algorithm, hence we do not use the usual dataset made of 180 samples.
- find the point that maximizes the EIF. If the EIF value at this point is sufficiently small, stop.
- evaluate the objective function at the point where the EIF is maximized. Update the Gaussian process model using this new point. Return to the previous step.

The EGO optimization represents the most interesting algorithm to compare with the newly developed POD-based approaches, as it embeds the concept of adaptivity and trade-off between design space exploration and surrogate model exploitation.

- **AFPGA1, AFPGA2, AFPGA3**: the surrogate model employed is the same as **FPGA3**, but the training method is different and an adaptive sampling strategy is added. In particular, we decided to follow a different approach: we want to check if, with a limited computational budget, we get better results by adaptively training the POD model. Hence, we split the surrogate training phase in three contributions: an *a-priori* contribution, sampling the design space with the LHS technique and producing  $M_{apr}$  samples; an iterative, adaptive sampling aimed at improving the modal basis and enriching the ensemble dataset with  $M_{adp}$  samples; a series of  $M_{opt}$  genetic optimizations, each producing an optimal candidate to

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---

update the ensemble and recompute the surrogate. The last phase will be also called real-time updating. The three strategies differ for the relative amount of these three contributions as highlighted in table 6.1, keeping fixed the total computational budget. The POD energy content is 99%. The snapshot size  $N$  is 201488;

- **AMPGA1**: the surrogate model employed is the SM2 model already studied in chapter 5. The FOM/ROM interface is defined at  $d = 0.35$  chord length from the airfoil leading edge and the FOM domain is depicted with blue colors in figure 5.8. However, the training method is different as it embeds *a-priori*, auto-adaptive and optimal samples as described earlier. The POD energy content is 99%. The snapshot size  $N$  is 91792;
- **AMPGA2**: the surrogate model employed is the SM3 model already studied in chapter 5 and used for **MPGA1,MPGA2** optimizations. The FOM/ROM interface is defined at  $d = 1.25$  chord length from the airfoil leading edge and the FOM domain is depicted with orange colors in figure 5.8. However, the training method is different as it embeds *a-priori*, auto-adaptive and optimal samples as described earlier. The POD energy content is 99%. The snapshot size  $N$  is 75232;

The optimization set up is the same for all the approaches, except for the last two. A population of 64 individuals is let evolve for 150 generations with a 80% crossover rate and a 2% mutation rate every time a new optimal sample has to be added to the ensemble. Hence, a total number of 9600 evaluations are required for each optimization process. The set up of AMPGA1 and AMPGA2 slightly differ as the adopted surrogate models are slightly more expensive, as reported in figure 5.14. In order to increase the frequency of model updating stages, a population of 48 individuals is let evolve for just 10 generations and the process is repeated 48 times to iteratively provide new optimal samples. The new feature is that each optimization step is a restart of the previous one with re-evaluation of the population candidates as the surrogate model has been updated meanwhile. In other words, the idea is to update the surrogate model more frequently (after just 10 GA generations instead of 150) even if with smaller amounts of improvement (10 generations are not enough to converge the GA).

By looking at the details of the SBO approaches described so far, it seems quite natural to divide them in two main classes: the non-adaptive (FPGAx,MPGAx), i.e. those

without any adaptation/real-time updating, and the adaptive optimizations (KGA, EGO, AFPGA<sub>x</sub>, AMPGA<sub>x</sub>). Consequently, the presentation of the obtained results will follow this logical sequence.

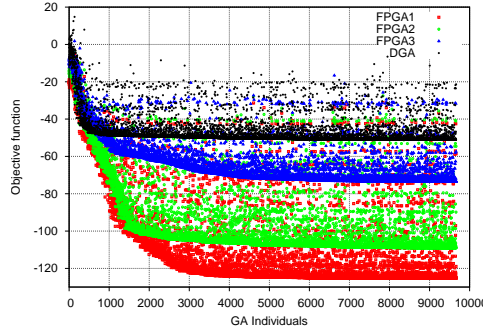
## 6.4 Non-adaptive optimization results

Figure 6.2-(a,b) shows the convergence history of the three FPGA optimizations compared to the plain DGA (black circles) on the left (full GA history) and the two MPGA optimization history on the right. In the latter figure, only the convergence history of the best candidates for each generation are reported instead of the whole optimization history. Another important point is that, while the DGA predictions (black circles and black line) are obtained with the CFD solver, the POD-based predictions (green, red and blue circles, blue and red lines) are the surrogate ones. For example, the red circles do not indicate that FPGA1 reached objective levels significantly better than DGA, but simply that the predicted values of the airfoil performances have been strongly overestimated. The plot clearly highlights that, whatever the energy content, the full-POD approximation is not able to match the “true” data during the search process. Moreover, the general FPGA trend is to strongly underestimate the CFD prediction in terms of objective function evaluation. On the other hand, the MPGA model agreement with the CFD progress is very satisfying, both in terms of trends and accuracy. Figure 6.2-(c,d), 6.2-(e,f) and 6.2-(g,h) confirm these results as they show the convergence history of some design variables (leading edge radius, upper  $A_4$  and lower  $A_1$ ) as a function of the progressing generations. Again, MPGAs behave quite well. The plots in figure 6.2 propose an interesting feature: the FPGA optimizations push the lower  $A_1$  variable towards the lower bound (-0.1), while the “truth” and MPGA converge towards the variable upper boundary at 0.

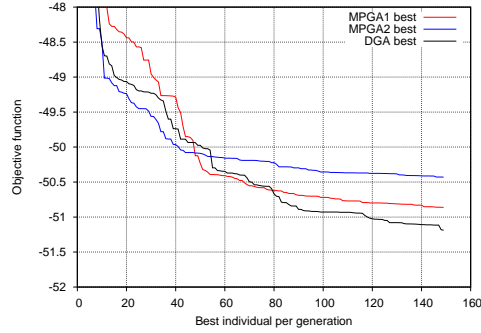
## 6.5 Adaptive optimization results

Figure 6.3 shows the convergence history of the iterative SBSO KGA run. After about 6-7 SBO iterations, the Kriging model has been improved enough to predict very closely to the CFD solver, as reported in the right-hand figure: it shows the Kriging prediction compared to the true one for the optimal candidate that is sequentially added to the

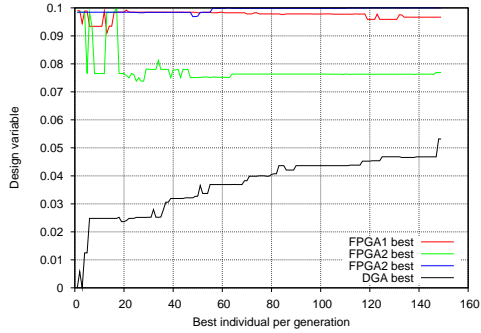
## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW



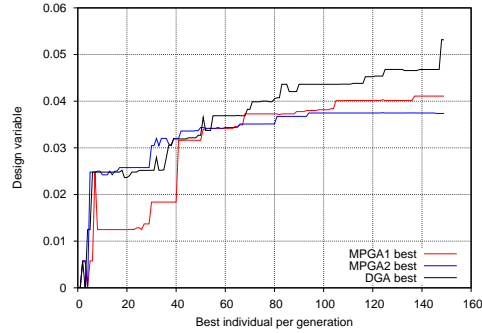
(a) FPGA, surrogate objective function



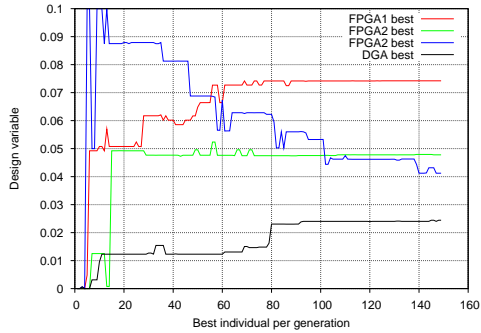
(b) MPGA, surrogate objective function



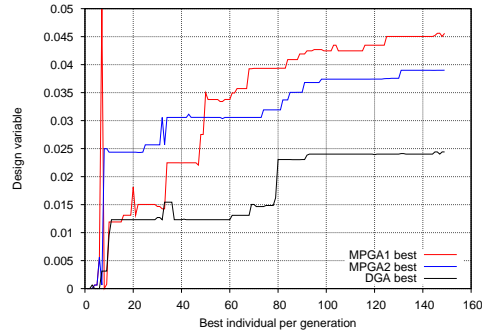
(c) FPGA, nose radius design variable



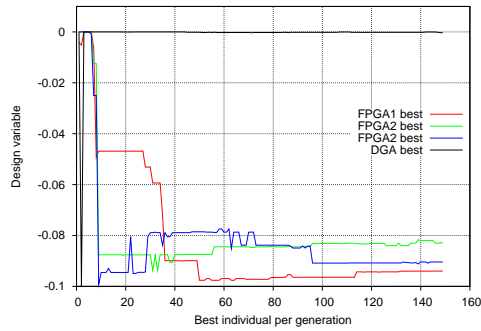
(d) MPGA, nose radius design variable



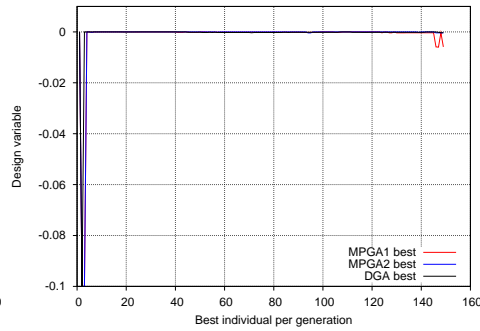
(e) FPGA, design variable upper  $A_4$



(f) MPGA, design variable upper  $A_4$



(g) FPGA, design variable lower  $A_1$

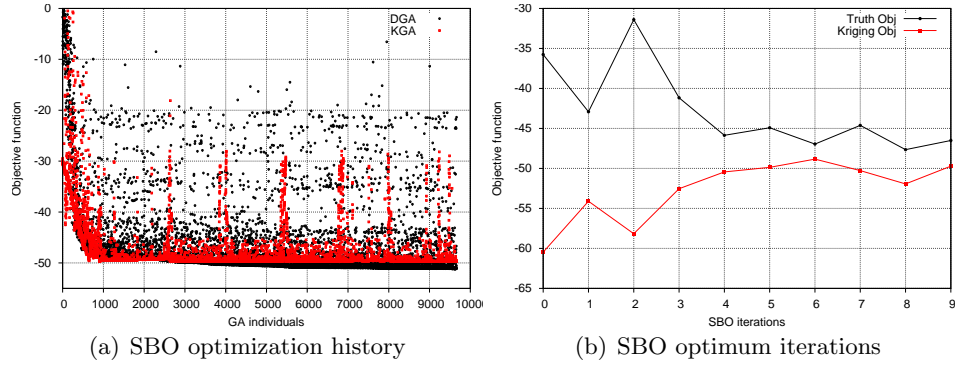


(h) MPGA, design variable lower  $A_1$

**Figure 6.2:** Non-adaptive POD-driven optimization history



original database after each surrogate-based optimization. The convergence history on the left-hand figure corresponds to the final SBO iteration, once the Kriging model has been optimized. The 9<sup>th</sup> iteration candidate will be considered as the KGA optimum, as it shows the lowest truth objective function among the SBO minimizers.

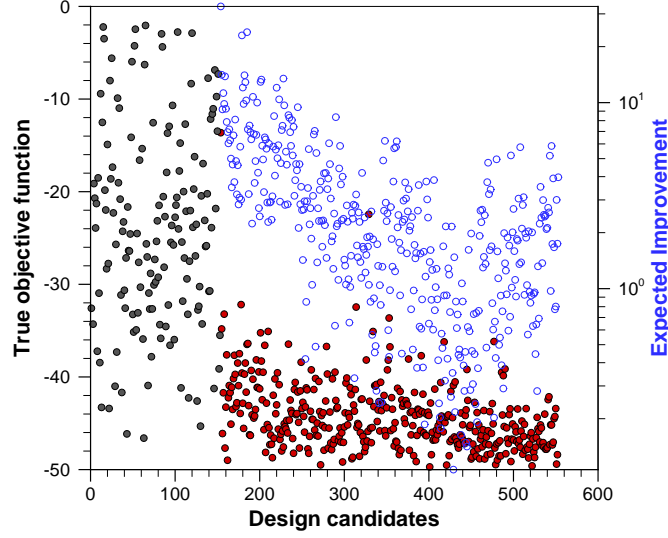


**Figure 6.3:** Kriging-based optimization history

Figure 6.4 reports the convergence history of the EGO optimization. Grey circles depict the initial DOE sampling (153 candidates), while red circles denote the candidate with minimum expected improvement found at each iteration. The graph also reports the expected improvement values in blue white-filled circles and logarithmic scale (right axis). It is clearly evident how the progressive decrease of the EIF produces a better quality of the Kriging model which in turn results in a minimization of the “true” objective function. The convergence history of the AFPGA1, AFPGA2 and AFPGA3 optimizations are reported in figure 6.5 together with the objective function computed on the training points. In the proposed plot, each point represents a single high-fidelity evaluation. The points connected with solid lines represents the sequence of optima computed by (1) updating the surrogate model and (2) running the GA optimization  $M_{opt}$  times. It is fairly evident how the adaptive sampling, which is triggered respectively after 32, 16 and 4 *a-priori* sampling, helps to find interesting design candidates even before the optimization stage. The best candidate of AFPGA1 is even found during the off-line adaptive DoE and not during the real-time optimization enrichment. This is due to the fact that, as the weight of the *a-priori* DoE is much bigger than the other cases (67% of the total off-line sampling compared to 33.3% for AFPGA2 up to 8.3% for AFPGA3), the ensemble is slightly enriched with an “intelligent” method

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---



**Figure 6.4:** EGO optimization convergence history

and it is unable to accurately explore the design space in the following phases. On the other hand, the auto-adaptive method works well for the same reason, i.e. because the margin of improvement is very high. Hence, the ratio  $\frac{M_{apr}}{M_{adp}}$  should be kept low. However, another important feature is related to the AFPGA3 method: indeed, it shows that, by lowering the ratio  $\frac{M_{apr}}{M_{adp}}$  too much (up to 0.09), the performance of the method deteriorates. These considerations give an helpful hint about the right combination of *a-priori* and adaptive sampling: the ratio  $\frac{M_{apr}}{M_{adp}}$  should be kept between 0.1 and 0.5.

This information is exploited in tuning the parameters for AMPGA1 and AMPGA2 optimizations. Figure 6.6 shows the “true” objective functions of the training samples and of the sequence of optima candidates. Even if the AMPGA1 performs quite well, it exhibits similar characteristics of the AFPGAx optimization. On the other hand, the AMPGA2 optimization reaches superior levels of performance as the computed optimum gets very close to the “truth” optimum, i.e. the DGA optimum. The following section will give details about the optima computed with each of the presented methodology.

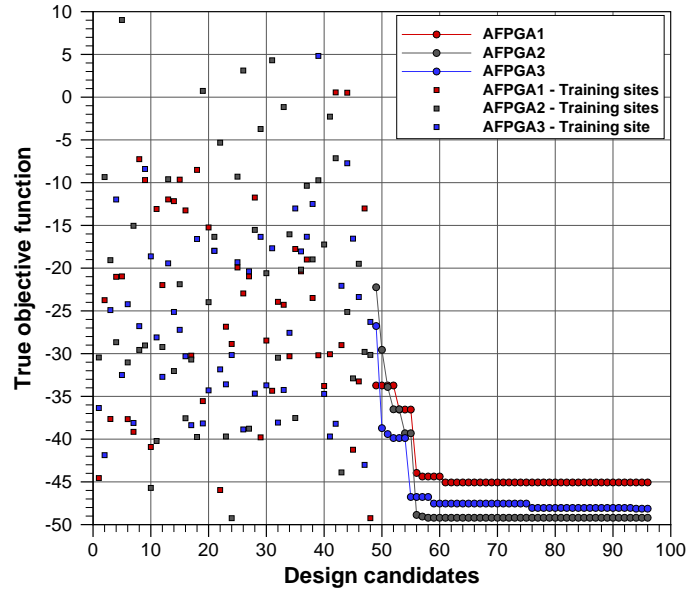


Figure 6.5: AFPGA optimizations convergence history

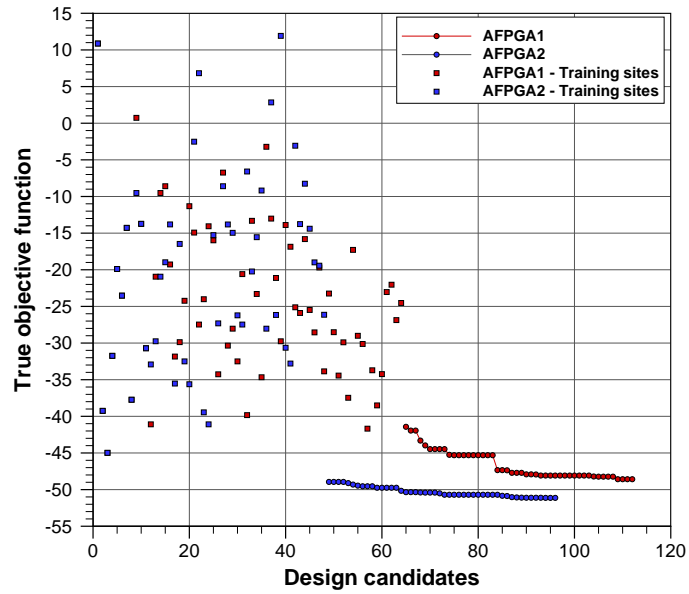


Figure 6.6: MFPGA optimizations convergence history

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---

### 6.6 Optima analysis

In the following, ten optimal candidates will be considered to assess the optimization results, namely the optima from run DGA, FPGA3, MPGA1, KGA, EGO, AFPGA1, AFPGA2, AFPGA3, AMPGA1 and AMPGA2. FPGA3 and MPGA1 have been selected among FPGAx and MPGAx optima because they are the closest to the high-fidelity DGA optimum. The objective function breakdown for each optimal candidate is summarized in table 6.2. The table reports both the “true” data, obtained by re-computing each design with the CFD solver, and the predicted objective function as calculated by the surrogate model. Each optimum does not satisfy the pitching mo-

Opt. run ID	Truth Obj	Predicted Obj	Penalty	$C_l$	$C_d$	$C_m$
DGA	-51.18	-51.18	1.025	0.619	0.0118	-0.0602
MPGA1	-48.70	-50.86	1.13	0.578	0.0116	-0.0606
FPGA3	-38.33	-73.45	0.608	0.553	0.0142	-0.0578
KGA	-47.65	-51.94	0.585	0.612	0.0127	-0.0576
EGO	-49.71	-49.71	0.530	0.618	0.0123	-0.0573
AFPGA1	-49.24	-47.14	1.12	0.635	0.0126	-0.0606
AFPGA2	-49.20	-52.61	0.551	0.631	0.0127	-0.0574
AFPGA3	-48.13	-47.88	1.29	0.583	0.0118	-0.0614
AMPGA1	-48.58	-44.61	0.567	0.576	0.0117	-0.0575
AMPGA2	-51.13	-50.31	0.947	0.612	0.0117	-0.0597

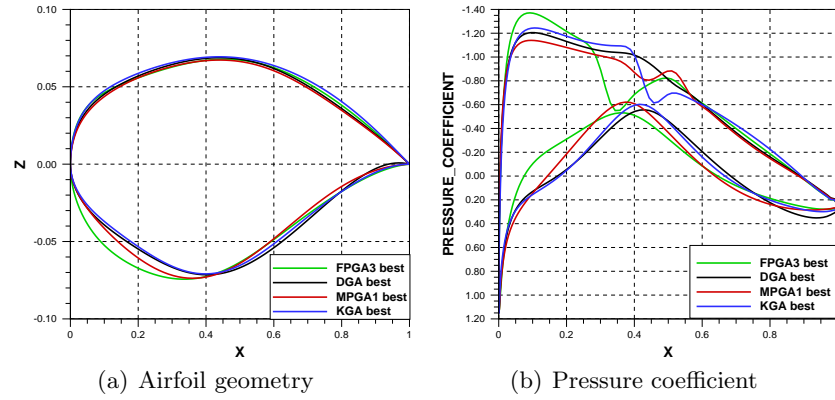
**Table 6.2:** Optimal candidates, obj. function breakdown

ment constraint because the quadratic penalty function and its weight, chosen in the problem definition, purposely do not enforce this constraint strictly to have a less stiff optimization problem. Indeed, getting precisely into the constraint boundaries would have probably penalized too much the aerodynamic efficiency, i.e. the actual objective function, while applying small penalties near a constraint boundary gives more flexibility to the search of the optimal design.

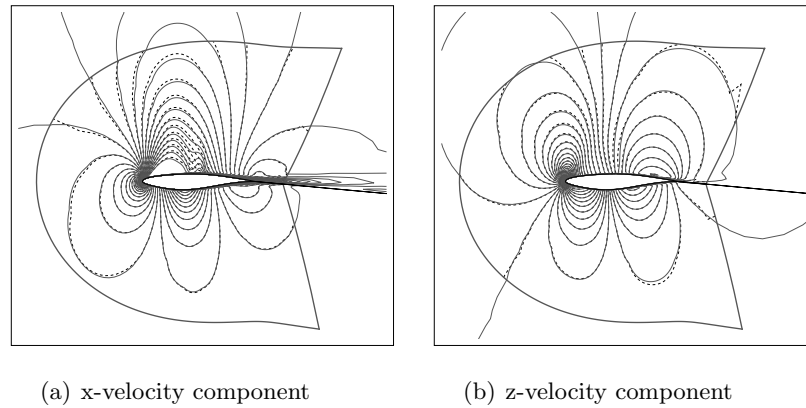
#### 6.6.1 Non-adaptive optima

Among the non-adaptive methods (KGA is here considered as non-adaptive to set a comparison), optimal design coming from MPGA1 and KGA are closer to the plain one in terms of global performance. MPGA1 optimum catches almost perfectly the DGA

constraint violation, while KGA design performs even better on pitching moment but at the cost of a slightly lower aerodynamic efficiency. FPGA3 design, although using 75 POD modes, does not belong to an optimal subset but exhibits a small penalty. A more interesting comparison is proposed in figure 6.7 where the optimal airfoils shape (left-hand) and pressure coefficient (right-hand) are depicted. It should be noted that the high-fidelity DGA optimization (black curves) is able to find a shock-less configuration by properly designing the upper airfoil side. The best surrogate solution



**Figure 6.7:** Non-adaptive optimal candidates comparison



**Figure 6.8:** MPGA1 optimum, velocity contour comparison (POD - dashed lines; CFD - solid lines)

is the MPGA1, where a weak shock appears on the suction side but at a lower lift level. The plots provide to give an insight explanation of why the FPGA3 candidate shows poor aerodynamic performances: the optimal leading edge radius, as also depicted in

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---

figure 6.2-(c,d), is almost twice the DGA value and this feature causes an over-expansion on the suction side which in turn makes the shock wave occur more upstream and stronger (green curve). Another significant design feature is observable on the Kriging-based best candidate (blue curve), with reduced rear loading to limit nose-down pitching moment and trim drag associated with the rear location of the center of pressure. This beneficial feature is counterbalanced by the lift production increase in the fore airfoil part and, consequently, by a more pronounced pressure jump across the shock wave. In figure 6.8 the velocity components contour are compared for MPGA1 optimum as obtained from full CFD (solid grey lines) and zonal CFD/POD (dashed black lines) computations. A general agreement can be noticed, even if some slight discrepancies on the CFD/POD boundary interface still exist which may deteriorate the high-fidelity prediction.

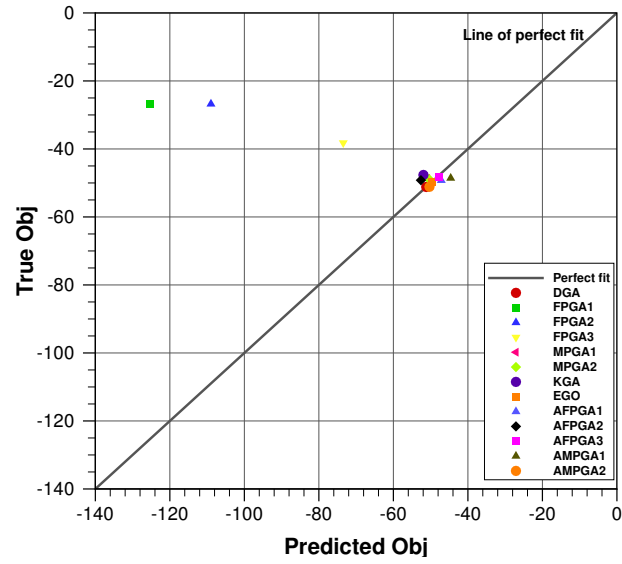
### 6.6.2 Adaptive optima

In order to highlight the path of improvements that has been followed, we report in figure 6.9 a correlation plot, representing both the non-adaptive and adaptive optima in the plane whose  $x$ -axis is the surrogate prediction, differing in nature from one optimum to another as different surrogates were used, and the  $y$ -axis is the “true” prediction, i.e. obtained with the ZEN CFD solver. Two different zooming levels are set, as they reflect the non-adaptive and adaptive process: indeed, the FPGA1, FPGA2 and FPGA3 optima show very large discrepancies between the two predictions, hence they will be located very far from the line of perfect fit. However, a trend is observable as, increasing the POD energy content (passing from FPGA1 to FPGA3), the best candidate gets closer to the true optimum. By looking at the top part of the figure, a clustering of the remaining optima is observable, so that a closer look is offered in the bottom figure for better understanding. Among the adaptive optima, the AMPGA2 and EGO method produce the best results and demonstrates the benefits of opportunely coupling the zonal approach and an “intelligent” design space sampling. Indeed, these optimal candidates are the closest to the target point in the sense of the Euclidean distance in the representation plane. The following figures 6.10, 6.11, 6.12, 6.13, 6.14 and 6.15 show some aerodynamic details of each candidate. Field Mach number contour maps and surface pressure/skin friction coefficients are reported to compare both compressible and boundary layer characteristics. In particular, it is clearly evident how the AMPGA2

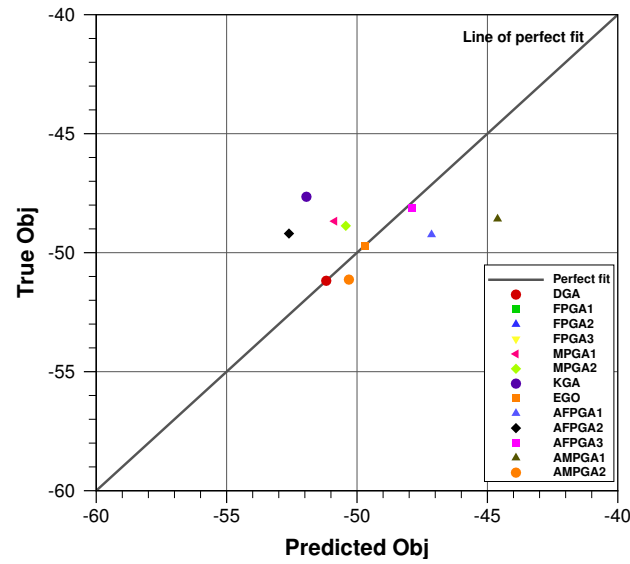
individual presents a shock-less behaviour as the true optimum, moreover with a more flat pressure profile and a less pronounced recompression on the suction side. This is a quite unique feature among the observed optimal airfoils.

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---



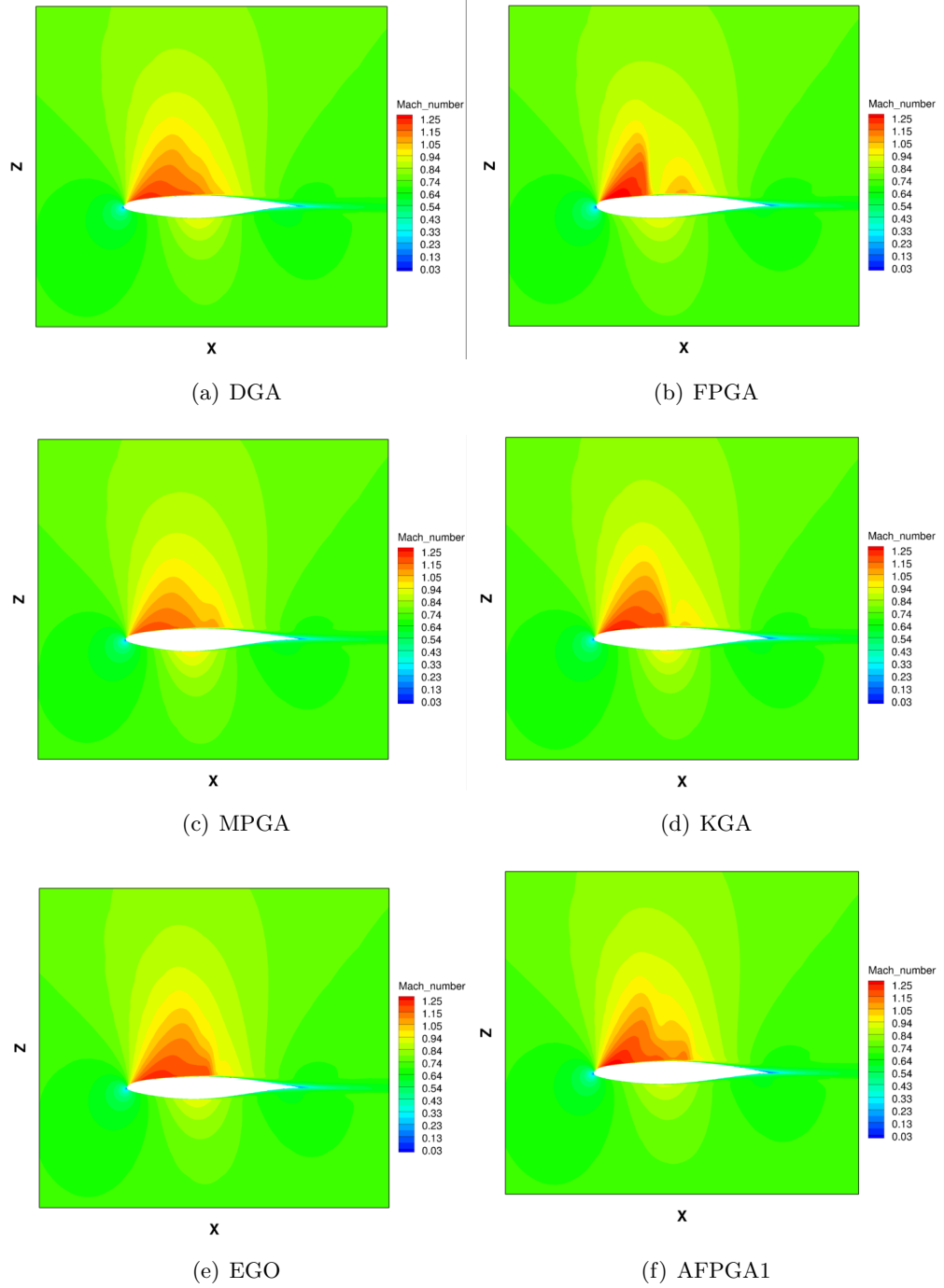
(a) Large view



(b) Zoom near the truth optimum

**Figure 6.9:** Computed optima in the surrogate vs truth objective plane

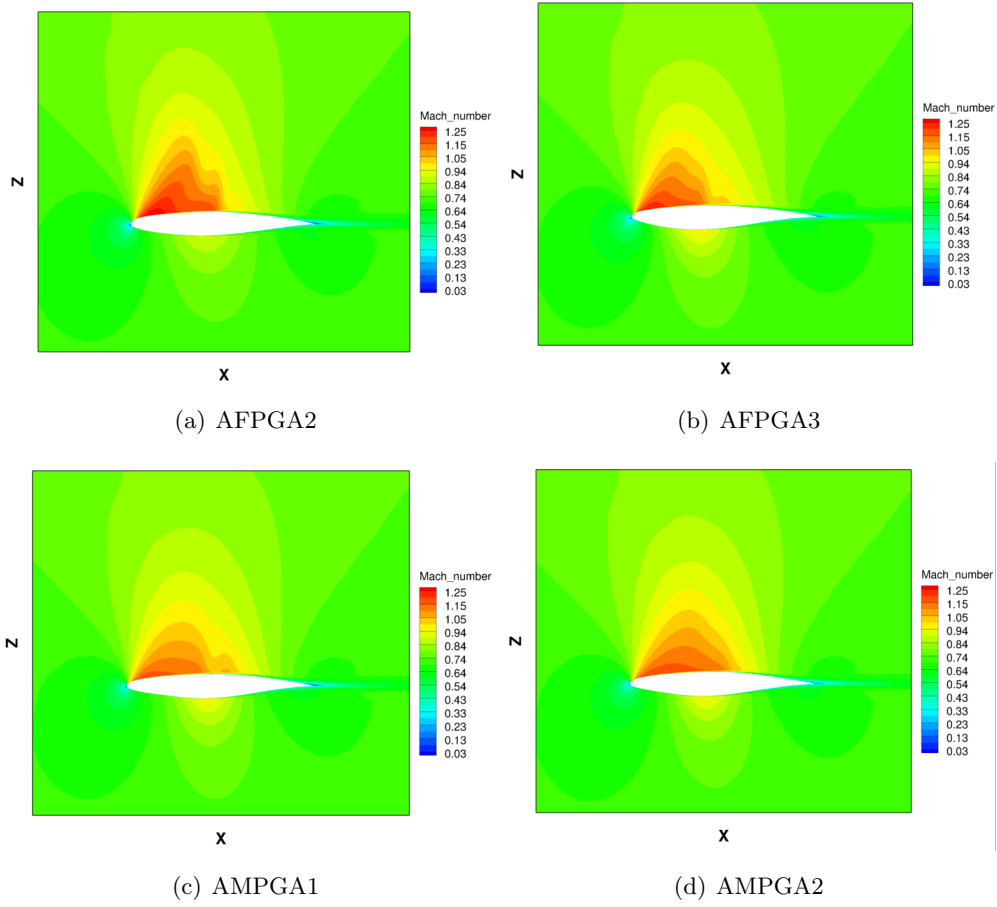




**Figure 6.10:** Optimal candidates, Mach number contour

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

---



**Figure 6.11:** Optimal candidates, Mach number contour

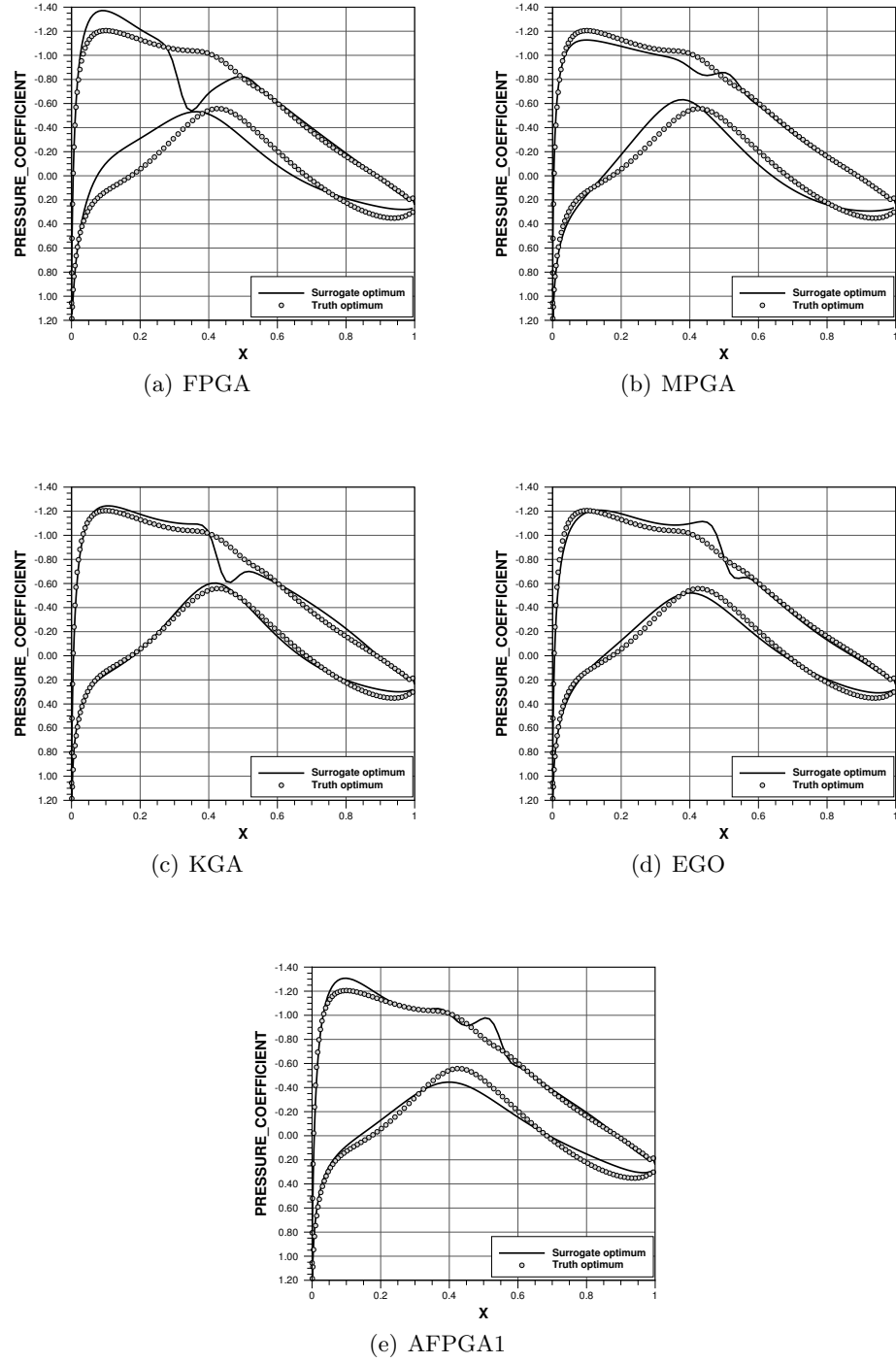
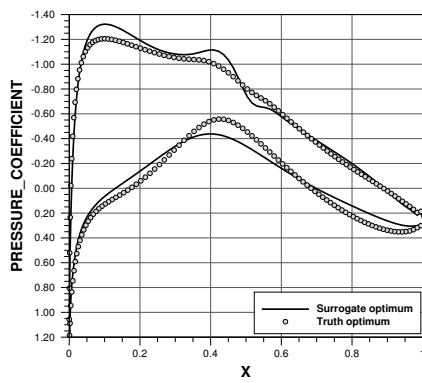


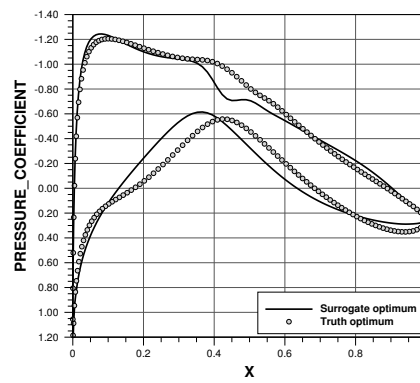
Figure 6.12: Optimal candidates, pressure coefficient distributions

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

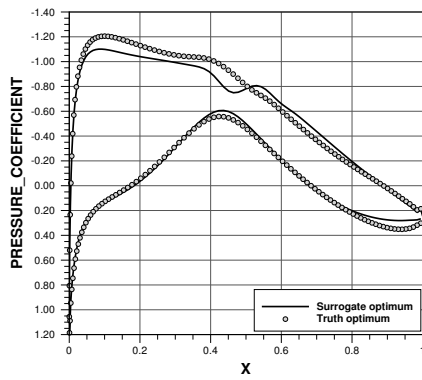
---



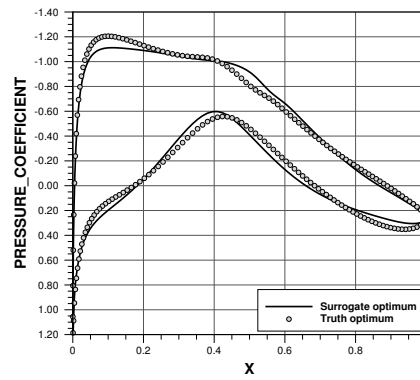
(a) AFPGA2



(b) AFPGA3

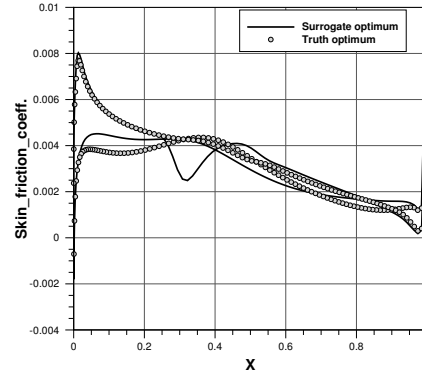


(c) AMPGA1

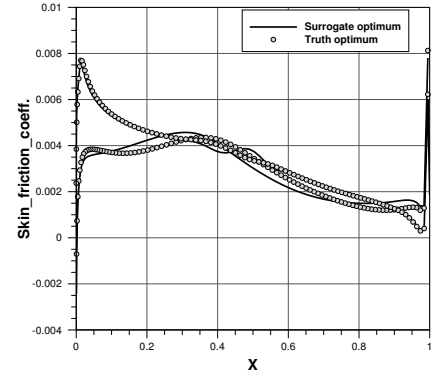


(d) AMPGA2

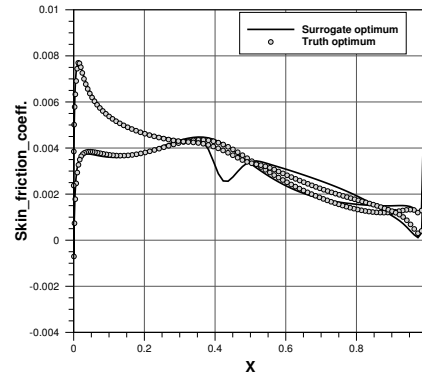
Figure 6.13: Optimal candidates, pressure coefficient distributions



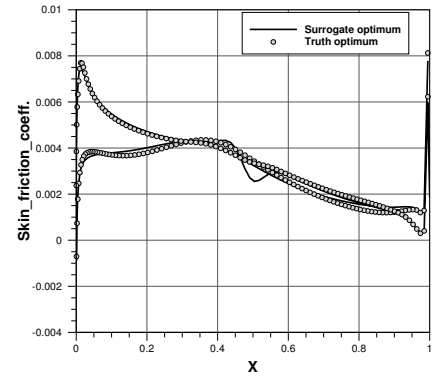
(a) FPGA



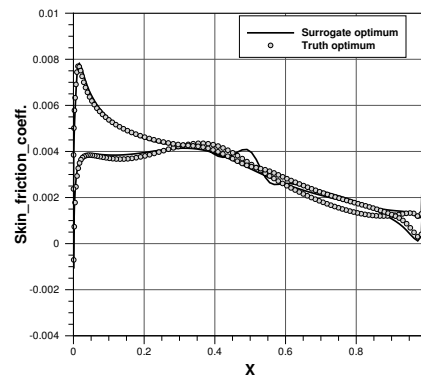
(b) MPGA



(c) KGA



(d) EGO

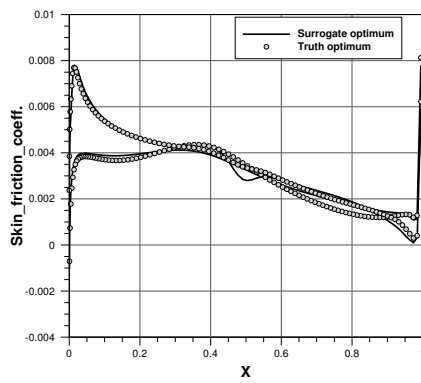


(e) AFPGA1

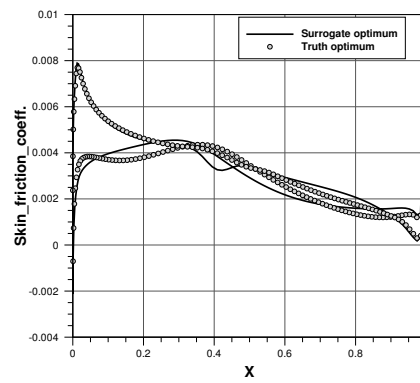
**Figure 6.14:** Optimal candidates, skin friction distributions

## 6. POD-BASED EVOLUTIONARY OPTIMIZATION IN TRANSONIC FLOW

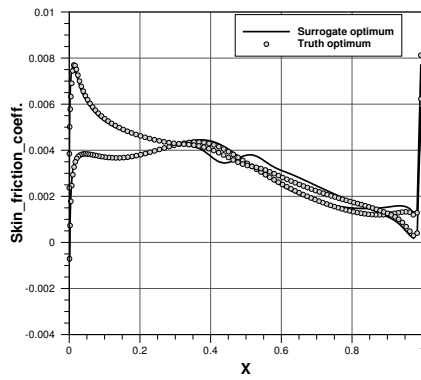
---



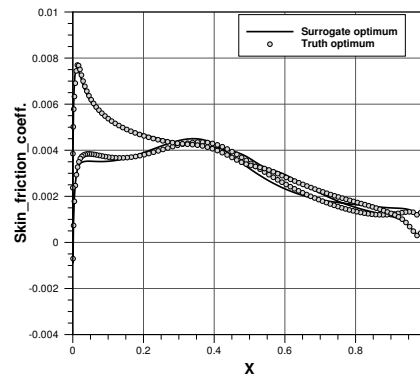
(a) AFPGA2



(b) AFPGA3



(c) AMPGA1



(d) AMPGA2

**Figure 6.15:** Optimal candidates, skin friction distributions

## Conclusions

The aim of the present research work was to investigate and study *ad-hoc* computational techniques to ease the solution of complex aerodynamic shape optimization problems such those commonly encountered in aerospace design at industrial level. Among the various approaches that are subject of research investigation, we chose to focus on *ad-hoc* surrogate methods. In particular, we demonstrated that the well-known Proper Orthogonal Decomposition approach is not adequate to provide reliable predictions in peculiar aerodynamic conditions like transonic flow and when the boundary of the computational domain changes like in shape optimization. We proposed a zonal approach to de-couple the strong non-linearities occurring near the body-wall from the POD approximation. This zonal approach proved to give reliable results at a reduced computational cost compared to the full CFD simulation. Furthermore, we showed that the zonal approach can give an accurate approximation of the true optimum when trained with specifically designed adaptive sampling techniques. The latter have been purposely conceived to improve the POD model machinery, namely the basis vectors and coefficients. By using such an “intelligent” design of experiment method, the high-fidelity computational budget can be further reduced and the overall performance of the design loop is increased. The beneficial effects of this approach has been illustrated by comparing several surrogate-based optimization processes on the shape design of a two-dimensional airfoil. The extension of the methodology to complex three-dimensional problems is straightforward and under way. Indeed, one of the main advantages of the proposed methodology is its relative insensitivity to the curse of dimensionality of the design parameter space. On the other hand, the larger snapshot size required by

## 7. CONCLUSIONS

---

three-dimensional CFD flow fields, where millions of unknowns may be handled, does not represent a big issue with current linear algebra numerical solver technology. Another significant advantage of the zonal approach with respect to other surrogates lies in its favourable scaling property when the third dimension is introduced because the ratio between CFD-solved and POD predicted points decreases. Furthermore, zonal POD allows to solve the high-fidelity flow field locally, i.e. only where it is required by geometry-driven considerations. This represents a tremendous benefit when the complexity of the design case grows. As an example, if the goal is to optimally fit a nacelle body in a already optimal wing, the high-fidelity computation zone can be restricted to catch only the wing-nacelle interaction phenomena, leaving to the POD model the prediction of the outside field. To further bridge the gap with real-world applications and needs, future works will be focused on validating the proposed methodology in a large-scale, multi-point aerodynamic problems involving huge design parameter spaces and aiming at predicting the aerodynamic characteristics in deep transonic flow.



# References

- [1] Jones, D. R. A taxonomy of global optimization methods based on response surfaces. *J. of Global Optimization* **21**, 345–383 December (2001).
- [2] <http://www.acare4europe.com/html/documentation.asp>.
- [3] Schrauf, G. Katnet, key aerodynamic technologies for aircraft performance improvement. In *Fifth Community Aeronautical Days 2006*.
- [4] Airbus. Global market forecast 2009-2028. [www.airbus.com](http://www.airbus.com).
- [5] Boeing-Commercial-Airplanes. Current market outlook 2009. [www.boeing.com](http://www.boeing.com).
- [6] Raymer, D. P. *Aircraft Design: A Conceptual Approach and Rds-student, Software for Aircraft Design, Sizing, and Performance Set (AIAA Education)*. AIAA (American Institute of Aeronautics & Ast, (2006).
- [7] Jones, D. R., Schonlau, M., and Welch, W. J. Efficient global optimization of expensive black-box functions. *J. of Global Optimization* **13**, 455–492 December (1998).
- [8] Schonlau, M., Welch, W. J., and Jones, D. R. Global versus Local Search in Constrained Optimization of Computer Models. *Lecture Notes-Monograph Series* **34** (1998).
- [9] Törn, A. and Zhilinskias, A. *Global optimization*. Springer-Verlag, Berlin ; New York :, (1989).
- [10] Alexandrov, N. M., Lewis, R. M., Gumbert, C. R., Green, L. L., and Newman, P. A. Optimization with variable-fidelity models applied to wing design. Technical report, (1999).

## REFERENCES

---

- [11] Booker, A. J., Dennis, J. E., Frank, P. D., Serafini, D. B., Torczon, V., and Trosset, M. W. A rigorous framework for optimization of expensive functions by surrogates. *Structural and Multidisciplinary Optimization* **17**, 1–13 (1999). 10.1007/BF01197708.
- [12] Gutmann, H.-M. A radial basis function method for global optimization. *J. of Global Optimization* **19**, 201–227 March (2001).
- [13] Queipo, N., Haftka, R., Shyy, W., Goel, T., Vaidyanathan, R., and Kevintucker, P. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences* **41**(1), 1–28 (2005).
- [14] Simpson, T. W., Toropov, V. V., Balabanov, V., and Viana, F. A. C. *Design and analysis of computer experiments in multidisciplinary design optimization: a review of how far we have come – or not*. Number AIAA 2008-5802. American Institute of Aeronautics and Astronautics, (2008).
- [15] Forrester, A. I. J. and Keane, A. J. Recent advances in surrogate-based optimization. *Progress in Aerospace Sciences* **45**(1-3), 50–79 (2009).
- [16] Sóbester, A., Leary, S., and Keane, A. A parallel updating scheme for approximating and optimizing high fidelity computer simulations. *Structural and Multidisciplinary Optimization* **27**, 371–383 (2004). 10.1007/s00158-004-0397-9.
- [17] Goel, T., Haftka, R. T., Shyy, W., and Queipo, N. V. Ensemble of surrogates. *Structural and Multidisciplinary Optimization* **33**(3), 199–216 (2007).
- [18] Mifsud, M. *Reduced-order modelling for high-speed aerial weapon aerodynamics*. PhD thesis, Cranfield University - College of Aeronautics, (2008).
- [19] Antoulas, A. C., Sorensen, D. C., and Gugercin, S. A survey of model reduction methods for large-scale systems. *Contemporary Mathematics* **280**, 193–219 (2001).
- [20] Loeve, M. *Probability theory / M. Loeve*. Springer-Verlag, New York :, 4th ed. edition, (1977).
- [21] Lumley, J. L. The Structure of Inhomogeneous Turbulent Flows. In Atmospheric turbulence and radio propagation, Yaglom, A. M. and Tatarski, V. I., editors, 166–178. Nauka, Moscow (1967).

## REFERENCES

---

- [22] LeGresley, P. and Alonso, J. Investigation of non-linear projection for pod based reduced order models for aerodynamics. In *39th AIAA Aerospace Sciences Meeting and Exhibit*, number 01-0926 (39th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2001).
- [23] B.I. and Epureanu. A parametric analysis of reduced order models of viscous flows in turbomachinery. *Journal of Fluids and Structures* **17**(7), 971 – 982 (2003).
- [24] Bui-Thanh, T., Damodaran, M., and Willcox, K. Proper orthogonal decomposition extensions for parametric applications in compressible aerodynamics. In *AIAA 2003-4213, 21st Applied Aerodynamics Conference* (, Orlando, Florida, 2003).
- [25] Bui-thanh, T., Damodaran, M., and Willcox, K. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA Journal* **42**(8), 1505–1516 (2004).
- [26] Sirovich, L. Turbulence and the dynamics of coherent structures. I - Coherent structures. II - Symmetries and transformations. III - Dynamics and scaling. *Quarterly of Applied Mathematics* **45**, 561–571 October (1987).
- [27] Holmes, P., Lumley, J., and Berkooz, G. *Turbulence, Coherent Structures, Dynamical Systems and Symmetry*. Cambridge University Press, Cambridge, (1996).
- [28] Dowell, E., Hall, K., Thomas, J., Florea, R., Epureanu, B., and Heeg, J. Reduced order models in unsteady aerodynamics. In *AIAA Paper 99-1261*, (1999).
- [29] Hall, K. C., Thomas, J. P., and Dowell, E. H. Reduced-order modeling of unsteady small-disturbance flows using a frequency-domain proper orthogonal decomposition technique. In *AIAA Paper 99-0655*, (1999).
- [30] Everson, R. and Sirovich, L. Karhunen–loève procedure for gappy data. *J. Opt. Soc. Am. A* **12**(8), 1657–1664 Aug (1995).
- [31] Tang, L., Chen, P., Liu, D., Gao, X., Shyy, W., Utturkar, Y., and Zhang, B. Proper orthogonal decomposition and response surface method for tps/rlv structural design and optimization: X-34 case study. In *43rd AIAA Aerospace Sciences Meeting and Exhibit* (, Reno, Nevada, 2005).

## REFERENCES

---

- [32] LeGresley, P. and Alonso, J. Dynamic domain decomposition and error correction for reduced order models. In *41st AIAA Aerospace Sciences Meeting and Exhibit*, number 03-0250 (41st AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2003).
- [33] Lucia, D. J., King, P. I., and Beran, P. S. Reduced order modeling of a two-dimensional flow with moving shocks. *Computers & Fluids* **32**(7), 917 – 938 (2003).
- [34] Lucia, D. J. *Reduced Order Modeling for High Speed Flows with moving shocks*. PhD thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, (2001).
- [35] Buffoni, M., Telib, H., and Iollo, A. Iterative methods for model reduction by domain decomposition. *Computers & Fluids* **38**(6), 1160 – 1167 (2009).
- [36] Toal, D. J. J., Bressloff, N. W., and Keane, A. J. Geometric filtration using pod for aerodynamic design optimization. In *Proceedings of 26th AIAA Applied Aerodynamics Conference*, 1–13, (2008).
- [37] Braconnier, T., Ferrier, M., Jouhaud, J.-C., Montagnac, M., and Sagaut, P. Towards an adaptive pod/svd surrogate model for aeronautic design. *Computers & Fluids* **40**(1), 195 – 209 (2011).
- [38] Lieu, T. and Farhat, C. Adaptation of pod-based aeroelastic roms for varying mach number and angle of attack: Application to a complete f-16 configuration. In *AIAA Paper 2005-7666, U.S. Air Force T&E Days*, (2005).
- [39] Lieu, T., Farhat, C., and Lesoinne, M. Pod-based aeroelastic analysis of a complete f-16 configuration: Rom adaptation and demonstration. In *AIAA Paper 2005-2295, 46th Structures, Structural dynamics & Materials Conference*, (2005).
- [40] Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, Massachusetts, January (1989).
- [41] Deb, K. Multi-objective genetic algorithms: Problem difficulties and construction of test problems. *Evolutionary Computation* **7**(3), 205–230 (1999).

- [42] Kulfan, B. M. Universal parametric geometry representation method. *Journal of Aircraft* **45**(1), 142–158 (2008).
- [43] <http://www.salome-platform.org/>.
- [44] Boerstoel, J. W., Spekreijse, S. P., and Vitagliano, P. L. The design of a system of codes for industrial calculations of flows around aircraft and other complex aerodynamic configurations. In *AIAA-92-2619-CP*, 207–229, (1992).
- [45] Spekreijse, S. P., Boerstoel, J. W., Vitagliano, P. L., and Kuyvenhoven, J. L. Domain modeling and grid generation for multi-block structured grids with application to aerodynamic and hydrodynamic configurations. In NASA CP-3143, Smith, R. E., editor, 207–229, (1992).
- [46] Amato, M. and Catalano, P. Non linear  $\kappa \varepsilon$  turbulence modeling for industrial applications. In *ICAS 2000 Congress* (, Harrogate, UK, 2000).
- [47] Amato, M. and Iaccarino, G. Mathematical Numerical Modelling for a Multiblock RANS flow solver. Technical report, Centro Italiano Ricerche Aerospaziali, Capua, Italy, October (1995). CIRA-TR-95-140.
- [48] Vicini, A. and Quagliarella, D. Inverse and direct airfoil design using a multiobjective genetic algorithm. *AIAA Journal* **35**(9), 1499–1505 September (1997).
- [49] Vicini, A. and Quagliarella, D. Airfoil and wing design through hybrid optimization strategies. *AIAA Journal* **37**(5), 634–641 May (1999).
- [50] Iuliano, E. and Quagliarella, D. Efficient aerodynamic optimization of a very light jet aircraft using evolutionary algorithms and rans flow models. In *IEEE Congress on Evolutionary Computation*, 1–10, (2010).
- [51] Iuliano, E. and Quagliarella, D. Efficient aerodynamic optimization of a very light jet aircraft using evolutionary algorithms and reynolds-averaged navier-stokes flow models. *Journal of Aerospace Engineering* **225**, 1109–1129 (2011).
- [52] Drela, M. Newton solution of coupled viscous/inviscid multielement airfoil flows. In *AIAA, Fluid Dynamics, Plasma Dynamics and Lasers Conference* (American Institute of Aeronautics and Astronautics (AIAA), Seattle, WA, 1990). AIAA Paper 90-1470.

## REFERENCES

---

- [53] Cella, U., Quagliarella, D., and Donelli, R. Design and optimization of a transonic natural laminar flow airfoil. In *Atti dell' XVIII Congresso Nazionale* (Associazione Italiana di Aeronautica e Astronautica (AIDAA), Volterra, Italy, 2005).
- [54] [http://en.wikipedia.org/wiki/gray\\_code](http://en.wikipedia.org/wiki/gray_code).
- [55] ESDU. *Computer program for estimation of lift curve to maximum lift for wing-fuselage combinations with high-lift devices at low speeds*, ESDU 99031, oct (1999).
- [56] Chandrashekarappa, P. and Duvigneau, R. Radial Basis Functions and Kriging Meta-models for Aerodynamic Optimization. Rapport de recherche RR-6151, INRIA, (2007).
- [57] Cook, P., Firmin, M., McDonald, M., and Establishment, R. A. *Aerofoil RAE 2822: pressure distributions, and boundary layer and wake measurements*. Technical memorandum / Royal Aircraft Establishment. RAE, (1977).
- [58] Pettit, C. and Beran, P. Reduced-order modelling for flutter prediction. In *AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, number 00-1446 in AIAA paper. AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, (2000).
- [59] Montgomery, D. C. *Design and Analysis of Experiments*. John Wiley & Sons, (2006).
- [60] Iuliano, E. Towards a pod-based surrogate model for cfd optimization. In *Proceedings of the Eccomas CFD & Optimization Conference* (, Antalya, Turkey, 2011).
- [61] Goblet, J. and Lepot, I. Two adaptive doe strategies for pod-based surrogate models. Technical report, ROM&O Project Report, CleanSky JTI-GRA, Call for Proposal JTI-CS-2009-1-GRA-05-004, Proposal number 255779, (2010).
- [62] Rippa, S. An algorithm for selecting a good value for the parameter  $\epsilon$  in radial basis function interpolation. *Advances in Computational Mathematics* **11**, 193–210 (1999). 10.1023/A:1018975909870.
- [63] Iuliano, E. and Quagliarella, D. Surrogate-based aerodynamic optimization via a zonal pod model. In *Proceedings of the EUROGEN 2011 Conference* (, Capua, Italy, 2011).

## REFERENCES

---

- [64] Eldred, M., Bichon, B., Adams, B., and Mahadevan, S. *Structural Design Optimization Considering Uncertainties*, volume 1 of *Structures and Infrastructures series*, chapter Overview of Reliability Analysis and Design Capabilities in DAKOTA with Application to Shape Optimization of MEMS, 401–432. Taylor & Francis Group (2008).
- [65] Eddy, J. E. and Lewis, K. Effective generation of pareto sets using genetic programming. In *Proceedings of ASME Design Engineering Technical Conference*, (2001).